NAVAL POSTGRADUATE SCHOOL

Monterey, California

THESIS

COST ESTIMATION OF CONTRACTOR PROVIDED SUPPLY SUPPORT FOR AVIATION SIMULATORS

by

Edward Milton Biggers

June 1991

Thesis Advisor: Dan C. Boger

Approved for public release; distribution is unlimited

Unclassified
SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1a. REPORT SECURITY CLASSIFICATION
UNCLASSIFIED

1b. RESTRICTIVE MARKINGS

2a. SECURITY CLASSIFICATION AUTHORITY

2b. DECLASSIFICATION/DOWNGRADING SCHEDULE

3. DISTRIBUTION/AVAILABILITY OF REPORT

4. PERFORMING ORGANIZATION REPORT NUMBER(S)

5. MONITORING ORGANIZATION REPORT NUMBER(S)

6a. NAME OF PERFORMING ORGANIZATION
Naval Postgraduate School

6b. OFFICE SYMBOL
OR

7a. NAME OF MONITORING ORGANIZATION

6c. ADDRESS (City, State, and ZIP Code)
Monterey, CA 93943-5000

7b. ADDRESS (City, State, and ZIP Code)

8a. NAME OF FUNDING/SPONSORING ORGANIZATION

8b. OFFICE SYMBOL

9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER

8c. ADDRESS (City, State, and ZIP Code)

10. SOURCE OF FUNDING NUMBERS

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.	WORK UNIT ACCESSION NO.

11. TITLE (Including Security Classification)
Cost Estimation of Contractor Provided Supply Support for Aviation Simulators

12 PERSONAL AUTHOR(S)
Edward Milton Biggers

13 TYPE OF REPORT
Master's Thesis

13b. TIME COVERED
FROM TO

14. DATE OF REPORT (Year, Month, Day)
1991, June

15. Page Count
70

16. SUPPLEMENTAL NOTATION
The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government

17. COSATI CODES

FIELD	GROUP	SUB-GROUP

18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
aircraft simulator, maintenance, supply support, cost estimating relationship, simulation, optimization

19. ABSTRACT (Continue on reverse if necessary and identify by block number)
This study develops cost estimating relationships (CER) to estimate the cost of supply support for the maintenance of aircraft simulators. These CER's will be used in the Contractor Operation and Maintenance of Simulators (COMS) environment. The analysis techniques used to develop and validate these relationships include linear regression (simple and multiple), analysis of univariate distributions, simulators modeling, and linear optimization modeling. The regression analysis concludes that no useful CER is present in the cost elements represented by the sample of data gathered from actual simulator operations. There is, however, a useful CER present in another smaller set of data derived from the successful bids of eight contractors. These submissions were for maintenance with and without supply support. A simulation model was constructed to provide an independent cost estimate for use with the maintenance CER developed above. Analysis of univariate distributions was used to transform maintenance data from an operational simulator suite for use in the verification of the simulation model. Finally, an optimization model formulation is recommended for further investigation to determine the best mix of contractor-provided and Navy-provided spare parts to complete the optimization of Navy expenditures for supply support for aviation simulators.

20 DISTRIBUTION/AVAILABILTIY OF ABSTRACT
X UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC

1a. REPORT SECURITY CLASSIFICATION
Unclassified

22a. NAME OF RESPONSIBLE INDIVIDUAL
Dan C. Boger

22b. TELEPHONE (Include Area Code)
(408)646-2607

22c. OFFICE SYMBOL
OR/Bo

DD Form 1473, JUN 86 Previous editions are obsolete. SECURITY CLASSIFICATION OF THIS PAGE
S/N 0102-LF-014-6603 Unclassified

Approved for public release; distribution is unlimited

Cost Estimation of Contractor Provided Supply Support for Aviation Simulators

by

Edward Milton Biggers
Commander, United States Navy
BBA, The University of Texas at Austin, 1969

Submitted in partial fulfillment
of the requirements for the degree

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
June 1991

Author: Edward Milton Biggers
Edward Milton Biggers

Approved by: Dan C. Boger
Dan C. Boger, Thesis Advisor

W. Max Woods
W. Max Woods, Second Reader

P. Purdue.
Peter Purdue, Chairman
Department of Operations Research

TABLE OF CONTENTS

ABSTRACT

This study develops cost estimating relationships (CER) to estimate the cost of supply support for the maintenance of aircraft simulators. These CER's will be used in the Contractor Operation and Maintenance of Simulators (COMS) environment. The analysis techniques used to develop and validate these relationships include linear regression (simple and multiple), analysis of univariate distributions, simulation modeling, and linear optimization modeling. The regression analysis concludes that no useful CER is present in the cost elements represented by the sample of data gathered from actual simulator operations. There is, however, a useful CER present in another smaller set of data derived from the successful bids of eight contractors. These submissions were for maintenance with and without supply support. A simulation model was constructed to provide an independent cost estimate for use with the maintenance CER developed above. Analysis of univariate distributions was used to transform maintenance data from an operational simulator suite for use in the verification of the simulation model. Finally, an optimization model formulation is recommended for further investigation to determine the best mix of contractor-provided and Navy-provided spare parts to complete the optimization of Navy expenditures for supply support for aviation simulators.

I. INTRODUCTION

A. BACKGROUND

In 1969 all aviation simulators were maintained and operated by personnel with a Tradevman (TD) rating. The Navy supply system was responsible for providing all repair parts and replacement assemblies for simulators. This system provided an adequate number of operational ready (OR) simulator hours to meet the needs of the various aviation communities throughout the Navy.

In the early 1980's, however, the Department of the Navy dramatically changed the way aviation simulators were maintained and operated. The Defense Officer Personnel Management Act (DOPMA), force level constraints and the goal to increase the size of the Navy to 600 ships were the major factors that prompted the changes. In order to man 600 ships, many new "at sea" billets would be needed, and the Navy end strength was already at the maximum level permitted by law. Decisions by the Congress forced these additional billets to be generated internally. The additional "at sea" manpower was generated by replacing much of the military-manned permanent shore services with civilian commercial contracted services.

The maintenance and operation of simulators became a prime target for conversion to commercial contract, and the concept

of Contractor Operation and Maintenance of Simulators (COMS) was born.

The conversion to COMS started in earnest in 1984 and was virtually complete by 1985 when the TD rating was disestablished. The original contracts were for maintenance and operation only, with the Navy continuing to be responsible for supply support. The COMS contracts were written so that the contractor was not charged down time when the Navy was not able to supply the required repair parts.

The maintenance of a complete stock of repair parts is difficult for the Naval supply system. The large number of parts unique to simulators, coupled with their low utilization rates, places an undue burden on a system which is not designed to handle parts on a small scale. The DOD acquisition regulations further hamper efforts to have sufficient spare parts to support the COMS contractor. The requirement for competition, or waiver for sole source acquisition, and the lead time required by the contracting process itself, frequently cause unusually high prices and unacceptable delays in the arrival of vital repair parts. The effect of these delays can cause an excessive number of unusable trainers and lead to decreased readiness for Naval aviation units.

One solution to this dilemma may be a new generation COMS contract that makes the contractor responsible for some level of supply support. To aid in the conduct of a feasibility

study to support the implementation of this new COMS contract, it is necessary to devise methods to answer the following questions:

1. How much does it cost to provide supply support?
2. What is the cost of a COMS contract?
3. Is there a best mix of contractor-supplied or Navy-supplied spare parts?

B. SCOPE AND LIMITATIONS

1. Scope

This research focused on the development of two CERs that can be used to estimate the cost of spare parts and the cost of a COMS contract required to maintain a simulator for one year.

This thesis summarizes the analysis and presents the results. Additional ideas are presented that outline the procedures needed to determine the best mix of Navy furnished/contractor furnished spare parts.

2. Limitations

The accuracy of the results of this analytical process is no better than that of the data used. Further limitations exist because of the small number of available simulators with useful data for this project. Other specific limitations will be discussed throughout this thesis as they occur.

The intent of this thesis is to demonstrate analytical procedures that lead to the determination of a CER. The accuracy of the resulting CERs is not expected to be

sufficient to allow its use in the contracting or budgeting process.

II. COST ESTIMATION

> A cost estimate is a judgment or opinion regarding the cost of an object, commodity, or service. This judgment or opinion may be arrived at formally or informally by a variety of methods, all of which are based on the assumption that experience is a reliable guide to the future." [Ref.1:p.1]

Cost estimation techniques are diverse and range from intuition to detailed application of labor and engineering cost standards. The five techniques most prevalent in The Department of Defense are industrial engineering standards; rates, factors, and catalog prices; cost estimating relationships; specific analogies; and expert opinion. There are many more techniques in the civilian sector. The determination of the technique to be applied is dependent on the data available, what the estimate is to be used for, the time available to do the study, and other factors that affect the direction of the study. For the purposes of this thesis, cost estimating relationships and industrial engineering standards will be the only two methods explored.

A. COST ESTIMATING RELATIONSHIPS

Cost estimating relationships (CERs) are equations derived from statistical analysis that are used to estimate the cost of a product based on some characteristic of that product. This method is preferred because of its ease of application and accuracy. Its use requires the availability of high

quality data and the existence of relationships among elements of that data base. A CER can be generated using regression analysis processes. The details of that process are discussed in the next chapter.

B. INDUSTRIAL ENGINEERING STANDARDS

On some occassions sufficient data are not available, or a useful CER can not be derived from the data, or a change in the process or materials makes historical data unusable. In these instances industrial engineering standards must be used.

> Estimating by industrial engineering procedures can be broadly defined as an examination of separate segments of work at a low level of detail and a synthesis of the many detailed estimates into a total. [Ref. 1:p.2]

In this estimation scheme the process of production is broken down into its smallest elements. The cost of labor and material for each of these elements of production is then computed and totaled. The results are then multiplied by factors, determined by management, to estimate overhead and other indirect costs. The summation of these direct and indirect costs equals the total cost estimate for the product.

C. PROBLEMS

As is the case with all estimation processes, there are some inherent problems in each of these procedures. In the case of cost estimating relationships, there are two limitations. The first is the inherent uncertainty always associated with the use of statistics. The second is the

uncertainty of the application of a particular CER to a certain situation.

The industrial engineering standards technique has three drawbacks. First, it takes a large number of personnel with a detailed knowledge of the production process to compute the production element costs. Second, small mistakes in the production element costs can be magnified when multiplied by the predetermined factors in the computation of indirect costs. Third, the whole cost frequently turns out to be greater than the sum of all the production element costs.

In considering the benefit to be derived from a cost estimation, one must always keep in mind that the estimations must be reasonable and structurally sound. The usefulness of the cost estimation may also be affected by the analyst himself. The lack of personal knowledge or familiarity with the systems being considered or a personal bias interjected into the analysis by the analyst can detract from the credibility of the estimate.

III. METHODS OF ANALYSIS

A. DETERMINING COST ESTIMATING RELATIONSHIPS

1. Simple Regression Analysis

Simple regression analysis is defined and explained in books that discuss data analysis. In Bachelder's An Introduction to Equipment Cost Estimating [Ref. 1], simple linear regression is defined as a simple two-variable model which describes the linear relationship between the two variables by the equation

$$y = a + bx + e \qquad (3.1)$$

where y represents the dependent variable (cost in this case), x represents the independent variable (any of the other data elements), and e is a random error or fluctuation. The constants a and b represent the Y-axis intercept and the slope of the line, respectively. Equation (3.1) represents the linear relationship between x and y, and it can be used to estimate y if an accurate value is available for x. The parameters a and b are estimated from data pairs (x_i, y_i)...(x_n, y_n) using well established analytical methods. The equations for these estimates A and B for a and b, respectively, are as follows:

$$A = \overline{y} - B\overline{x}$$

$$B = \frac{\sum (x_i - \overline{x})\ (y_i - \overline{y})}{\sum (x_i - \overline{x})^2}$$

when
$$\overline{x} = \frac{1}{n}\sum y_2,\ \overline{y} = \frac{1}{n}\sum y_2$$

The STSC program Statgraphics was used in this thesis to perform the statistical analyses. This program uses the ordinary least squares method to estimate the values of a and b, and will perform the calculations need to make statistical inferences about the parameters a and b and the line a + bx.

a. <u>Standard Error of the Regression (SE)</u>

The standard error of the regression is a value that represents the standard deviation of the estimated values of the error, e. The equation for SE in terms of the data pairs (x_i, y_i) is

$$SE = \sqrt{\sum (y_2 - (A + Bx_i))^2}$$

where A and B are the least squares estimate of a and b, respectively.

b. T-value

The t-value associated with the parameter b is defined as

$$t=\frac{B}{SE/\Sigma(x_i-\bar{x})^2} \qquad 3.2$$

It is used to test the significance of the relationship between x and y. This is done by performing a statistical test of the hypothesis that the slope coefficient, b, is equal to 0. That is, it is decided that $b \neq 0$ when the value of the t-statistic in equation 3.2 is larger in absolute value than the value of a number, $t_{\alpha,\ n-2}$, taken from standard student-t tables. The value of α is the level of significance for this test of hypothsis.

c. R-squared Value

The R-squared value is a measure of how well the regression equation predicts the value of y. It is frequently called the coefficient of determination, and equals one if all of the observed y_i values are on the predicted line of the estimated model, A + Bx.

2. Multiple Regression

If the regression of one variable on another fails to produce an adequate CER, then a technique called multiple regression can be used. This method is similar to simple regression except it has more than one independent variable. In the resulting representative equation

$$y = a + bx + cz + e, \tag{3.3}$$

y is still the dependent variable, x and z are independent variables, e is the random error or fluctuation, a is the Y-axis intercept, and the coefficients b and c represent the change in y for each change of one unit of their respective variables (x and z).

It is important to remember that each added variable reduces the accuracy of the statistical inferences made about the paramters a, b, c and about the line itself. This becomes critical when working with small data sets as it can decrease the credibility of the analysis.

The computed t-value and R-squared numbers are used for the same purposes as in simple regression.

a. Collinearity

Collinearity exists when two of the variables in a multiple regression model have, between themselves, a linear relationship. An indication that this condition exists is the presence of a high R-squared value while one or more of the variables has a low t-value.

B. SIMULATION MODELING TECHNIQUES

If statistical analysis fails, another form of analysis known as simulation modeling may be used. Simulation modeling is the second technique to be utilized in this thesis.

One dictionary definition of the verb simulate is as follows: "to assume or have the appearance or characteristics of." [Ref. 2]

This would appear to be a reasonably accurate description of the technique. The key to a successful simulation, however, is not just the "appearance or characteristics," but rather the operation or behavior of the simulation model.

A useful working definition of simulation is the following:

> The process of designing a mathematical or logical model of a real system and then conducting computer-based experiments with the model to describe, explain, and predict the behavior of the real system. [Ref. 3:p. 5]

Utilization of this second technique requires the construction of a model of the aircraft simulators that simulates the maintenance process. This model utilizes parameters, unique to the performance of the system being investigated, to estimate the number of man-years required to maintain the system. This information is then used to determine the cost of a COMS contract for a year.

To validate this model, it is necessary to analyze historical data from a known system and use the analysis results to run the simulation. The results of the simulation are compared to reality. If the two are statistically equivalent, the model is validated and usable.

1. Analysis of Maintenance Data

The analysis of the maintenance data is fairly straightforward. First, it is necessary to obtain a data base that contains the time each failure occurred and the maintenance time it took to repair each discrepancy. The time of failure data is used to determine the inter-arrival times

(time between failures). The inter-arrival times and repair times are then analyzed to determine their probability distributions and the parameters of those distributions. These parameters are fed into the model which then simulates the maintenance process.

a. Histogram

The first step in analysis is to display a histogram of the data. A histogram is a graph that shows the number of occurrences of data points within a bin (range of values). The histogram will indicate the location (mean) and variance of the probability distribution associated with the data. It may also give additional indications about this probability distribution.

b. Probability Plot

The probability plot is the primary means of determining the distribution reflected in the data set. This plot compares the quantile of a data set with the theoretical quantile of a selected distribution. If the two plots are reasonably close, the probability distribution that generated the sample is said to be the same, statistically, as the theoretical distribution.

c. **Kolmogorov - Smirnov Test**

The Kolmogorov - Smirnov test is used to determine the goodness-of-fit of a sample distribution to a theoretical distribution. It does this by comparing the absolute maximum distance between the sample cumulative distribution function and the theoretical cumulative distribution function. The computed significance level is used to determine if the sample distribution is the same as the theoretical distribution.

IV. COST ESTIMATION MODELS

A. COST DATA

The data used in this thesis were made available by Fleet Aviation Specialized Operational Training Group, Atlantic Fleet (FASO). The data represent the annual costs incurred in the acquisition and operation of forty-six separate aviation simulators for FY-87. The data elements provided include the cost of device acquisition (ACQCOST), number of hours the simulator was programmed for use (PGMHRS), COMS contract cost (COMCOST), utilities cost (UTILCOST), spare parts cost (SUPPCOST), total cost (TOTLCOST), and the cost per hour to operate the simulator (HRLYCOST). TOTLCOST is the sum of COMCOST, UTILCOST, SUPPCOST, and overhead costs for contract compliance inspection personnel. HRLYCOST is TOTLCOST divided by PGMHRS.

Additional data on the cost of eight five-year COMS contracts bid with and without supply support were provided by Naval Training Systems Center (NTSC), Orlando, Florida.

The first step in the analysis of this data was to see if a CER existed that would estimate the cost of supply support.

B. DEVELOPMENT OF CER USING FASO DATA

1. Simple Regression

Using the cost data supplied by FASO, a simple regression of SUPPCOST against all the other data elements demonstrated the absence of a useful CER with any of them. For the purposes of comparison, the t-statistic for 45 degrees of freedom and level of significance $\alpha=.05$ is 2.414. A statistical summary of these regressions is provided in TABLE 4.1. Note that the t-value of the slope in all cases but PGMHRS is larger than the t-statistic indicating that they are all significant, implying that the b is not equal to 0. However, the R-squared values are all much lower than that required for a reasonable explanation of the dependent variable.

TABLE 4.1 SIMPLE REGRESSION OF INDEPENDENT VARIABLES AGAINST THE DEPENDENT VARIABLE SUPPCOST

INDEPENDENT VARIABLE	PARAMETER	T VALUE	R SQUARED
UTILCOST	SLOPE	4.88562	35.17%
HRLYCOST	SLOPE	7.68515	57.31%
TOTLCOST	SLOPE	9.61059	67.73%
ACQCOST	SLOPE	4.94474	35.72%
PGMHRS	SLOPE	.42723	.41%
COMCOST	SLOPE	3.60911	22.84%

2. Multiple Regression

With the failure of simple regression to produce a reasonably accurate CER, a backward stepwise multiple

regression technique was tried. Since SUPPCOST is a component of both TOTLCOST and HRLCOST, it would have a linear relationship with both of those variables. Therefore, in order to insure that collinearity was not a problem, TOTLCOST and HRLCOST were not considered during the multiple regressions.

TABLE 4.2 shows the results of the multiple regression of SUPPCOST against the remaining four variables. The first step yields little significance for any of the variables, and the R-squared value is very low at 34.21. The next step eliminates the variable with the lowest t-value (COMCOST) and does the regression process again. The results of this step are not much better. The significance of each of the variables is less, and the R-squared value does not change appreciably. These results indicate that these data do not exhibit the relationships required to produce a CER that can be used to estimate spare parts cost.

The failure of the data to produce a usable CER is not surprising. The funding for spare parts comes from operational funds. For this reason, the cost of supplies is a measure of the availability of money as well as the cost of replacement repair parts.

TABLE 4.2. MULTIPLE REGRESSION OF INDEPENDENT VARIABLES AGAINST THE DEPENDENT VARIABLE SUPPCOST

RUN 1

Model fitting results for: SUPPCOST

Independent variable	coefficient	std. error	t-value	sig.level
CONSTANT	28435.122998	2.039639E41	1.3941	0.1708
PGMHRS	-8.033911	6.666066	-1.2052	0.2350
ACQCOST	0.002245	0.001477	1.5196	0.1363
COMCOST	-0.049499	0.110375	-0.4485	0.6562
UTILCOST	0.270332	0.205072	1.3182	0.1947

R-SQ. (ADJ.) = 0.3421 SE= 38015.056957 MAE= 20951.79413
Previously: 0.0000 0.000000 0.000000
46 observations fitted.

RUN 2

Model fitting results for: SUPPCOST

Independent variable	coefficient	std. error	t-value	sig.level
CONSTANT	25588.820008	1.919855E4	1.3329	0.1898
PGMHRS	-7.789751	6.580307	-1.1838	0.2432
ACQCOST	0.002071	0.001412	1.4669	0.1499
UTILCOST	0.232652	0.185282	1.2557	0.2162

R-SQ. (ADJ.) = 0.3546 SE= 37651.779438 MAE= 20686.445263
Previously: 0.3421 38015.056957 20951.794135
46 observations fitted.

C. DEVELOPMENT OF CER USING NTSC DATA

The data provided by Naval Training Systems Center (NTSC) consists of the successful bid for each of eight different simulator maintenance contracts. These bids included the cost of five years of maintenance both with and without supply support making it most applicable to the questions at hand.

The results of a simple regression of COMS cost without supply support (COMWOSUP) as the independent variable and cost of supply support (SUPPLY) as the dependent variable were very good. The t-value of the slope was 9.88510, compared to a tabled value of 2.998 for 7 degrees of freedom with an alpha=.01, indicating the significance of the coefficient. The R-squared value is a respectable 94.21. This is probably not good enough for budgeting purposes, but it is accurate enough to be used for the comparison of contracting strategies. The resulting CER is:

$$\text{Cost of Supply Support} = -37012.8 + .1565 * \text{COMS Cost} + e \quad (4.1)$$

It should be noted that the cost for supply support and the COMS cost are for a full five year contract.

The problem with this CER is that it is now necessary to come up with a method to determine an independent estimate for the cost of a COMS contract before it can be useful.

D. DEVELOPMENT OF CER FOR COMS CONTRACT COST

The regression analysis procedures were repeated with COMCOST as the dependent variable. The results were similar to the results of the regression with SUPPCOST above. The t-values were good for some of the regressions, but the R-squared values were not nearly high enough to indicate a quality CER. The multiple regression for COMCOST was not successful either. Stepwise regression also produced low R-squared values, again indicating a less than desirable CER. Hence, no CER exists in the FASO data set for the cost of a COMS contract.

E. SUMMARY OF REGRESSION RESULTS

Regression has yielded a CER for the cost of supply support. That CER uses the cost of the COMS contract as an independent variable. Regression failed to produce a usable CER for the cost of a COMS contract, so another method must be utilized to provide that cost. In the next chapter, such an alternate procedure will be demonstrated.

V. SIMULATION MODEL

The failure of the previous data sets to produce a CER provides an opportunity to explore the modeling of engineering standards to provide a method of cost estimation.

> An old proverb states that a journey of a thousands miles begins with but a single step. To paraphrase, a simulation analysis begins with a single step and must proceed methodically through several steps if useful results are to be derived and implemented. But before beginning any significant journey, it is important to know something about the route and noteworthy milestones along the way.
>
> . . .
>
> Simulation analysis is a descriptive modeling technique. As such, it does not provide the explicit problem formulation and solution steps which are an integral part of optimization models, such as linear programming. Consequently, one must specify in some detail a procedure for the development and use of simulation models to assure successful outcomes from their application. [Ref. 3:pp. 14-15]

The process utilized in this thesis to insure accurate results consists of

1) A definition of the problem and understanding of the process to be simulated so a good model formulation can be constructed;

2) A search for data and analysis of that data to determine its characteristics;

3) Development of a model to simulate the maintenance process;

4) Verification and validation of the model's code, logic and results;

5) Utilizing simulation results to determine the cost of a simulator maintenance contract.

A. PROBLEM DEFINITION AND FORMULATION

> Problem formulation is, in a sense, the most important step in a simulation analysis. Appropriate solutions to inappropriately formulated problems cannot be achieved. But before a problem can be formulated, it must be identified or found. Problem finding is, in reality, choosing from among several problems which are competing for the same resources. Criteria for selection include technical economic, and political feasibility, and the perceived urgency for a solution. Problem selection can have a significant impact on the ultimate success of the analysis and the implementation of the results. [Ref. 3:pp 16-17]

In order to formulate this simulation model, it is first necessary to have an understanding of the process to be simulated and define the problem to be solved.

1. The Simulator Maintenance Process

There are two types of simulator maintenance, scheduled maintenance and nonscheduled maintenance. Scheduled maintenance is done as required by the preventive maintenance schedule, is not variable, and will therefore not be part of the model.

Nonscheduled maintenance occurs as the result of a component failure or system breakdown. These component failures or system breakdowns can be discovered and reported in one of three scenarios. First, maintenance personnel can find a discrepancy during scheduled maintenance and report it. Second, a pilot can find a discrepancy during the conduct of a training mission and not report it until the end of the trainer period. Finally, a pilot can find a discrepancy during the conduct of a training mission that degrades the performance of the simulator below that required for the

completion of the mission, and the pilot reports that discrepancy immediately. All three of these scenarios are called breakdown events and will be assumed to occur randomly.

After a discrepancy has been reported, it is assigned to a technician for repair. This repair takes a variable length of time to complete. The completion of the repair makes the simulator available for use and is called a repair event. The simulator is scheduled for use and the process starts over again.

2. Problem Statement

To answer the questions posed in this thesis, it is necessary to determine how many man hours are required to maintain an aviation simulator. This will be converted to the number of man years required for maintenance, and used to determine a cost estimate.

B. SIMULATION MODEL DEVELOPMENT

The problem of determining the number of man hours required to maintain an aviation simulator can be answered by developing a Fortran language computer model to simulate the maintenance of an aviation simulator suite. This model should be easily modifiable to fit simulator suites consisting of any number of elements. An element is a segment of a simulator suite designed for a specific training mission (i.e., operational flight trainer (OFT), weapons system trainer (WST), etc).

Such a model was developed for this thesis. It utilizes a main program to initialize the parameters and perform the simulation by calling a variety of subroutines. These subroutines are the initialization subroutine, the time advance subroutine, the breakdown event subroutine, the repair event subroutine, the report generator subroutine, and random variable generators for the exponential, uniform, and normal distributions. The failure and repair subroutines are constructed from a block of code with one block for each simulator element. This enables the model to be expanded or contracted to meet the specifications of any simulator suite. At the completion of the simulation, the report generator prints out the parameters used for the failure and repair distributions, the number of maintenance personnel, maximum queue length, element operational ready rate, number of maintenance actions that took longer than three hours to repair, simulation run time, and total number of maintenance actions that occurred during the simulation. A complete listing of the program is included in Appendix A.

1. How to Use the Model

a. The first step is to load the program into a computer with a Fortran compiler.

b. Modify the parameters for the failure and repair distributions to reflect those of the simulator suite being simulated. An analysis of these distributions for the example considered in this thesis is in the next section.

c. Modify the number of elements in the program to reflect the physical composition of the simulator suite of interest. For the purposes of this thesis, the eight elements, Operational Flight Trainer (OFT) 1 and 2, Weapons System Trainer (WST) 1 and 2, Positional Type Trainer Tactics (PTT) 1 and 2, and Positional Type Trainer Cockpit (PTC) 1 and 2 have been used.

d. Estimate the number of maintenance personnel required to maintain the simulator suite, and enter that number for the value of NMTPER. Then, determine the length of time the simulation is to cover and enter that value for FEL(17).

e. Compile and run the model.

f. Note the results of interest, in this case operational ready rate and maximum repair queue length, and modify the number of maintenance personnel up or down until the desired results are obtained.

g. Determine the number of maintenance personnel required to perform the contract. This is done by multiplying the number of maintenance personnel needed to meet mission requirements by 3 to cover a 3 shift 24 hour day, and adding the appropriate number of overhead personnel. Overhead personnel include, but are not limited to, site supervisor, supply clerk, maintenance administration clerk, personnel to perform the required preventive maintenance (PM), and safety observers as required by Navy safety regulations.

h. Multiply the number of man-years required to maintain the simulator by the NTSC-provided cost for acquiring one man-year on contract, $45,000, to get a cost estimate for the COMS contract.

C. DATA SEARCH AND ANALYSIS

Data used in determining the parameters to be utilized by this model were obtained from an actual P-3 simulator site. Failure rate data were obtained for a 227 day period of time, and time to repair data were obtained for a different 90 day period of time. The different periods of time for the two data sets were driven by the way the records were kept by the

maintenance contractor. These data were analyzed to determine if the failure rate and repair times demonstrated the behavior of a known probability distribution.

1. Analysis of Data

a. <u>Interarrival Time</u>

The periodic occurrence of an event such as the breakdown of an aviation simulator is best described in terms of the time between breakdown events. This period of time is called the interarrival time. The data used to determine the interarrival times were taken from the maintenance action forms (MAF) that were filled out by the person discovering the malfunction and submitted to maintenance personnel. Once the failure times were entered into a data bank, they were processed by a simple *Fortran* program to convert them into interarrival times.

The analysis of the resultant interarrival times was done utilizing the IBM program GRAFSTAT. The "Fitting Probability Distributions" option of the "Analysis of Univariate Distributions" section of the Grafstat menu was used. This option utilized the fitted histogram/density and probability plots to analyze the data and determine its parameters. It was initially anticipated that the interarrival times would be exponentially distributed.

The data used for this thesis turned out to be very ill behaved. The data did, in fact, fail to conform to any known distribution, and it defied initial attempts at

analysis. It was therefore necessary to attempt data transformations in order to get any usable information.

(1) Transformation. Transformation is the systematic modification of data to make it conform to a distribution. The most utilized of these modification procedures are the logarithmic and exponential transformations. The process is a simple one; the data is raised to a certain power, or its logarithm is determined, and the resulting data set is analyzed. The exponential transformation proved to be useful and is represented by equation 5.1,

$$Y^a = X. \tag{5.1}$$

Y is the data, a is the power to which the data is raised, and X is a random variable that can be fitted to some distribution.

Unfortunately, even after transformation, much of the data continued to defy analysis. Probability plots of the raw data showed that much of it was bimodal or lacked good discipline. Neither of these problems came as a total surprise because of the inherent characteristics of the failure process.

(2) Problems. A bimodal situation is created when data points on a data set come from two or more distributions. That is to say, in this case, that the failure rate of the main computer is different than that of the motion system, which is different from that of the cockpit indications, etc.

When all the types of failures are thrown into one set of numbers, it can create many problems for the analyst.

The data problems appear to be the result of the process by which the data were recorded. Many times a discrepancy will be discovered during a training evolution, but, since it does not degrade training, it is not recorded until the end of the exercise. Sometimes several discrepancies will be identified at essentially the same time (the completion of the trainer time period). There is also a high probability that several malfunctions were discovered by maintenance personnel and corrected without any paper work to record the malfunction or the repair. All of these types of data excursions cause the data to be ill behaved and increase the difficulty of analysis.

(3) Solution. To counter these problems, several algorithms for the performance of the required analysis were devised and tested. The one that produced the best results is described as follows.

The first step of this algorithm was to uniformly distribute all blocks of discrepancies obviously written at the end of a training evolution. This redistribution was made over the last half of a three hour training period. The resulting data were then analyzed as described above in two different ways. First, each element (OFT1, OFT2, etc.) was individually analyzed. Second, the

elements were combined by type (OFT, WST, etc.) and analyzed again.

The results of this analysis are shown in TABLE 5-1. For each element and type, the second column shows exponent that was used to transform the interarrival times. The third column shows the value of the parameter of an exponential distribution which was fit to the transformed interarrival times for that element and type. The fourth column shows the significance level, using a Kolmogorov-Smirnov statistic, of the test of the hypothesis that the empirical distribution of the transformed interarrival times fit a theoretical exponential distribution with the parameter shown in the column three.

TABLE 5.1 PARAMETRIC ANALYSIS OF THE INTERARRIVAL TIMES FOR SIMULATOR ELEMENTS

ELEMENT	TRANSFORMATION EXPONENT	EXPONENTIAL PARAMETER	K.-S. SIGNIFICANCE LEVEL
OFT1	.84	51.59	.9198
OFT2	.40	4.39	.3131
OFT	.88	18.52	.7591
WST1	.64	15.77	.8061
WST2	.46	5.67	.9039
WST	.70	22.64	.9057
PTT1	.54	8.86	.5918
PTT2	.62	11.15	.6599
PTT	.77	28.05	.9619
PTC1	.69	33.65	.8591
PTC2	.36	3.06	.1296
PTC	.73	21.99	.9256

The most significant results were then used in accordance with the following rules:

1) If analysis resulted in good, significant parameters for both elements of a type of trainer, those individual parameters were used in the simulation.

2) If analysis resulted in very significant results coming from only one element of a type, then that parameter was used for both elements in the simulation.

3) If the most significant results came from the combined elements of type data, the parameter for the combined data was used, as is, for both elements in the simulation.

In the analysis of the failure data, rule 2 was applied to OFT1 and OFT2 (OFT1 data being used). Rule 3 was applied to WST1, WST2, PTT1, PTT2, PTC1, and PTC2. The parameters used for interarrival times in the simulation are presented in TABLE 5.2. The effects of these choices of the parameters on the overall model will be shown in Section D below.

TABLE 5.2 INTERARRIVAL TIME PARAMETERS USED IN THE SIMULATION MODEL

ELEMENT	TRANSFORMATION EXPONENT	EXPONENTIAL PARAMETER
OFT1	.84	51.59
OFT2	.84	51.59
WST1	.70	22.64
WST2	.70	22.64
PTT1	.77	28.05
PTT2	.77	28.05
PTC1	.73	21.99
PTC2	.73	21.99

b. Maintenance Repair Time

The length of time it takes a maintenance technician to repair an aviation simulator system discrepancy is called the maintenance repair time. The maintenance repair times are taken directly from the portion of the MAF that is filled out by the technician after he has finished the repair process.

The analysis of the maintenance repair time data was conducted identically to that of the interarrival time data, and the results were similar. The data turned out to be equally as ill behaved and the transformation process was again required to obtain usable results.

The decision process was the same as well, and TABLE 5.3 illustrates the transformation exponents and mean service time parameters used in the model.

TABLE 5.3 MEAN SERVICE TIME PARAMETERS USED IN THE SIMULATION MODEL

ELEMENT	TRANSFORMATION EXPONENT	MEAN SERVICE TIME	STANDARD DEVIATION
OFT1	.33	1.14	.22
OFT2	NATURAL LOG	2.08	.36
WST1	.32	1.46	.43
WST2	.32	1.46	.43
PTT1	.50	2.23	.74
PTT2	.50	2.23	.74
PTC1	.14	1.05	.07
PTC2	.14	1.05	.07

D. VERIFICATION AND VALIDATION OF THE MODEL

> Model verification and validation actually is concerned with three models: a conceptual model, a logical model, and a computer model. ...Verification focuses on internal consistency of a model, whole validation is concerned with the correspondence between the model and reality. [Ref. 3:p. 27]

Verification of the model was a continuing process throughout its development. As they were identified, code and logic errors were rectified until the model ran satisfactorily.

To validate the simulation model, the parameters derived by the analysis of available maintenance data were inserted into the simulation model. The model was set to run for 6000 hours which represents the number of hours of operation provided for in one year of operation (364 days - 104 weekend days - 10 govt holidays * 24 hours/day = 6000 hours). The program was run for two maintenance personnel (minimum number of personnel required by safety regulations for maintenance of energized equipment with high voltage components) and increased by one person until the maximum queue size was 0. The detailed results are shown in Appendix B.

It appears that four maintenance personnel are required to keep the simulators up without any delay time due to non-availability of maintenance personnel. The preventive maintenance (PM) manuals indicate that it takes 3.15 people to do the required PM.

All the information required to determine the number of maintenance personnel to maintain this simulator suit is

now known. That number is 18.15. TABLE 5.4 illustrates the determination of this number.

The simulator in question was being maintained by 18 people at the time the data was gathered. The model therefore appears to be validated as a good simulation of aviation simulator maintenance.

TABLE 5.4 NUMBER OF PERSONNEL REQUIRED TO MAINTAIN THE P-3 SIMULATOR MODELED

BILLET	NUMBER REQUIRED
SIGHT SUPERVISOR	1
SUPPLY CLERK	1
MAINTENANCE ADMINISTRATION CLERK	1
PM PERSONNEL	3.15
MAINTENANCE PERSONNEL	12
TOTAL	18.15

E. UTILIZATION OF MODEL RESULTS

The solution to the problem of determining the cost of maintenance and supply support for aviation simulators is at hand. Simple computations are all that is required to estimate those costs.

The cost of a contract requiring 18.15 men to maintain simulators is computed by multiplying 18.15 by $45,000 (the approved estimate for the cost of one man year of an average simulator technician). That cost is $816,750 for one year of maintenance service. This is within 5% of the cost of one year of service as delivered by the present contractor.

Now the developed COMS cost can be used in the developed CER (equation 4.1) to determine the cost for contractor supply support for this simulator suite. This cost is estimated to be $127,089 per year. This also compares favorably with the cost of the contract now in effect.

Therefore, one method of determining the cost of supply support provided by a COMS contractor has been determined.

VI. AREAS FOR FURTHER RESEARCH

Presently the contractor supplied/Navy supplied mix for spare parts is determined by what is in the simulator supply room at the time the contractor takes over maintenance of the simulator. An inventory is conducted, and the contractor is responsible for maintenance of that inventory until the completion of the contract period. While this is a step in the right direction, there are ways of determining a more cost efficient spare parts mix that considers all non trainer peculiar parts.

The best way to determine the most cost efficient parts mix is to construct a linear optimization program that minimizes the cost of parts support. A proposed basic formulation for that optimization program is:

INDEX

i = PART NUMBER

PARAMETERS

CNA = COST FOR NAVY ACQUISITION OF PART

CNS = COST FOR NAVY TO STORE PART

CNH = COST FOR NAVY TO HANDLE PART

CCA = COST FOR CONTRACTOR ACQUISITION OF PART

CCH = COST FOR CONTRACTOR TO HANDLE PART

NR = NUMBER OF PARTS REQUIRED FOR LIFE CYCLE OF SIMULATOR

VARIABLES

X_i = NUMBER OF PARTS $_i$ NAVY TO BUY

Y_i = NUMBER OF PARTS $_i$ CONTRACTOR TO BUY

FORMULATION

MINIMIZE

$$\Sigma_i \; CNA_i*X_i + CNS_i*X_i + CNH_i*X_i + CCA_i*Y_i + CCH_i*Y_i$$

SUBJECT TO

$$X_i + Y_i = NR_i$$

This basic formulation is simplistic by design. It has no constraints, other than the parts constraint, to allow for tailoring to any specific situation. For example, if there is a limited amount of operating funds (O&MN) the following constraint can be added:

$$\Sigma_i \; CNA_i*X_i \leq \text{FUNDS AVAILABLE}$$

Likewise, if there is a paucity of funds available for contractor supply support, the following constraint can be added:

$$\Sigma_i \; CCA_i*Y_i + CCH_i*Y_i \leq \text{FUNDS AVAILABLE}$$

Values for X_i can be assigned for those parts that are trainer peculiar and have to be purchased by the Navy at the time of acquisition of the simulator.

The ability to tailor this formulation to any simulator suite is limited only by reality. If a constraint can be identified and linearly represented, it can be included in the

formulation. The only real limitations are the availability of data and the format of that data.

Data were not available to support this model at the time the research was done for this thesis. The hard copy data that was acquired for a small subsystem did not have complete cost data for over-the-counter (contractor) purchase of most of the parts. A recommended format for the data to be optimized by this formulation is:

PART INDEX NUMBER	NSN	COST ACQUISITION NAVY		COST HANDLING NAVY		COST STORE NAVY
		ACQUISITION NAVY	ACQUISITION CNTR	HANDLING NAVY	HANDLING CNTR	STORE NAVY
1	1234567890	145.00	128.00	1.50	12.50	13.40

Once the data have been put into the right format, the formulation and its associated data can be an input into any matrix generator/linear solver. The resultant solution of the linear set of equations will provide the optimal mix of contractor/Navy acquisition of spare parts

Unfortunately, the absence of usable data prevents the verification of this optimization model. However, the formulation's simplicity, coupled with its sound theoretical foundation, would lend an air of credibility to the results obtained.

VII. SUMMARY AND RECOMMENDATIONS

A. SUMMARY

The results of the analysis in this thesis show that the aircraft simulator cost elements studied do not correlate well enough to allow for successful linear regression analysis. This is most likely caused by pressure applied to the level of spending on repair parts from sources other than the maintenance process. Examples are requirements to expend all funding during the quarter received and the necessity to transfer funds between commands to meet priority requirements in other areas.

However, it has been shown that there is a predictable relationship between the cost of a maintenance contract and the cost for providing supply support for that contract. This relationship is derived from a relatively small set of data drawn from eight successful bidders for simulator maintenance contracts. The accuracy of this model is excellent, with a high R-squared value (94.21). Utilization of this model will provide reasonably accurate government estimates for comparison with the bids from commercial contractors.

A maintenance process simulation model was created to provide an independent estimate for the cost of maintaining a simulator for one year. Its accuracy was validated by analyzing the performance of an actual simulator and utilizing

those performance parameters in the simulation model. The results were impressive. In addition to its accuracy, the model, because of its modular construction, is easily adaptable to any specific simulator suite.

This thesis has also provided a formulation for the determination of the optimal mix of contractor-supplied versus Navy-supplied repair parts. Even though it has not been verified or validated, this optimization formulation has great potential for savings if the time is taken to put the cost data in the format required.

B. RECOMMENDATIONS

The results of this study suggest that regression analysis and computer modeling can be of significant value in the estimation of supply support costs and maintenance costs for aircraft simulators. Continued utilization of these analysis techniques to update the CERs will further refine the accuracy of the results obtained.

The potential cost savings that can be gained by utilization of the optimization formulation developed herein cannot be overemphasized. The fervent pursuit of a usable data base for all simulators in the inventory is highly recommended and will pay dividends in the future.

APPENDIX A

THSMOD

```
*******************************************************************************************

                                        THSMOD

A simulation model written to simulate the maintenance requirements for an aviation simulator suit
and print out the results.

                                      WRITTEN BY

                                CDR EDWARD M. BIGGERS

                            TO SATISFY THE REQUIREMENTS FOR

                              A MASTER OF SCIENCE DEGREE IN

                                  OPERATIONS RESEARCH

                                  SEPTEMBER 15, 1989


        DEFINITION OF VARIABLES


        BRKDN       -       Break down subroutine
        CHKIN       -       Time break down occurred
        CHKINQ      -       Time break down added to the queue
        CLOCK       -       Time keeper for simulation
        ELAWMT      -       Simulator element waiting for maintenance
        EXPON       -       Exponential random variate generator
        F           -       Number of bread downs lasting longer that 3 hours
        FEL         -       Future events list
        IAT         -       Interarrival time
        IMEVT       -       Imminent event
        INITLZ      -       Initialize subroutine
        J,K,L       -       Do-loop index
        MITOF1      -       Mean interarrival time OFT1
        MITOF2      -       Mean interarrival time OFT2
        MITWS1      -       Mean interarrival time WST1
        MITWS2      -       Mean interarrival time WST2
        MITPT1      -       Mean interarrival time PTT1
        MITPT2      -       Mean interarrival time PTT2
        MITPC1      -       Mean interarrival time PTC1
        MITPC2      -       Mean interarrival time PTC2
        MQ          -       Maximum length of queue
        MSTOF1      -       Mean service time OFT1
        MSTOF2      -       Mean service time OFT2
        MSTWS1      -       Mean service time WST1
        MSTWS2      -       Mean service time WST2
        MSTPT1      -       Mean service time PTT1
        MSTPT2      -       Mean service time PTT2
*******************************************************************************************
*******************************************************************************************
        MSTPC1      -       Mean service time PTC1
        MSTPC2      -       Mean service time PTC2
        NAWTMT      -       Number of simulator elements waiting maintenance
        NMTBUS      -       Number of maintenance personnel busy
        NMTPER      -       Number of maintenance personnel
        NORML       -       Normal random variate generator
        NUMEVS      -       Number of events
        OPRDY1      -       Operational ready rate OFT1
        OPRDY2      -       Operational ready rate OFT2
        OPRDY3      -       Operational ready rate WST1
        OPRDY4      -       Operational ready rate WST2
```

OPRDY5	-	Operational ready rate PTT1
OPRDY6	-	Operational ready rate PTT2
OPRDY7	-	Operational ready rate PTC1
OPRDY8	-	Operational ready rate PTC2
REPAIR	-	Repair subroutine
RPTGEN	-	Report generator subroutine
SGMOF1	-	Standard deviation OFT1 service time
SGMOF2	-	Standard deviation OFT2 service time
SGMWS1	-	Standard deviation WST1 service time
SGMWS2	-	Standard deviation WST2 service time
SGMPT1	-	Standard deviation PTT1 service time
SGMPT2	-	Standard deviation PTT2 service time
SGMPC1	-	Standard deviation PTC1 service time
SGMPC2	-	Standard deviation PTC2 service time
SVT	-	Service time
TIMADV	-	Time advance subroutine
TNREP	-	Total number of repair events
TREPTM	-	Total repair time
TTMINQ	-	Total time in queue

To utilize this model, first determine the parameters for each of the elements of the simulator. These parameters include interarrival time, mean service time for repairs, and the standard deviation for those repair times. Insert these values into the main program. Modify the break-down/repair subroutines to emulate the number of elements in the simulator suit being modeled. Determine the number of events and enter that number into the main program.

Estimate the number of maintenance personnel required to maintain the simulator at the desired operational ready rate and enter that value in the main program.

Run the simulation program and note the results.

Modify the number of maintenance personnel until the operational ready rate is within an acceptable range, until the maximum queue size is 0, or until any other required condition is met.
****

```
      PROGRAM THSMOD
*****************************************************************************************
      MAIN PROGRAM
        1) INITIALIZE MODEL
        2) CALLS TIME-ADVANCE AND EVENT ROUTINES
        3) CALLS REPORT GENERATOR TO TERMINATE SIMULATION NORMALLY

*****************************************************************************************
      INTEGER F,ELAWMT(100),TNREP
      REAL MITOF1,MITOF2,MITWS1,MITWS2,MITPT1,MITPT2,MITPC1,MITPC2,
     1  MSTOF1,MSTOF2,MSTWS1,MSTWS2,MSTPT1,MSTPT2,MSTPC1,MSTPC2,
     2  FEL(17),CHKIN(8),CHKINQ(100),NORML,TREPTM(8),TTMINQ(8)
        COMMON /SIM/ MITOF1,MITOF2,MITWS1,MITWS2,MITPT1,MITPT2,MITPCl,
     1  MITPC2,MSTOF1,MSTOF2,MSTWSl,MSTWS2,MSTPT1,MSTPT2,MSTPC1,MSTPC2,
     2  SGMOF1,SGMOF2,SGMWS1,SGMWS2,SGMPT1,SGMPT2,SGMPC1,SGMPC2,NAWTMT,
     3  NMTBUS,CHKIN,CHKINQ,TREPTM,MQ,F,TNREP,NMTPER,ELAWMT,TTMINQ
        COMMON /TIMEKP/ CLOCK,IMEVT,NUMEVS,FEL
        NUMEVS = 17
*****************************************************************************************
          ASSIGN VALUES TO INPUT PARAMETERS
      (THESE VALUES COULD BE STORED IN A FILE AND READ INTO THE PROGRAM)
*****************************************************************************************
      MITOF1 = 25.799
      MITOF2 = 25.799
      MITWS1 = 22.64
      MITWS2 = 22.64
      MITPT1 = 28.053
      MITPT2 = 28.053
      MITPCl = 21.99
      MITPC2 = 21.99
      MSTOF1 =  1.14
      MSTOF2 =  2.0841
      MSTWS1 =  1.4604
      MSTWS2 =  1.4604
      MSTPT1 =  2.2272
      MSTPT2 =  2.2272
      MSTPCI =  1.051
      MSTPC2 =  1.051
      SGMOF1 =  0.2183
      SGMOF2 =  0.35516
      SGMWSl =  0.43201
      SGMWS2 =  0.43201
      SGMPT1 =  0.74425
      SGMPT2 =  0.74425
      SGMPCl =  0.073798
      SGMPC2 =  0.073798
      NMTPER =  2
*****************************************************************************************
    CALL INITIALIZATION ROUTINE
*****************************************************************************************
          CALL INITLZ
*****************************************************************************************
     CALL TIME-ADVANCE ROUTINE TO DETERMINE IMMINENT EVENT AND
       ADVANCE CLOCK TO THE IMMINENT EVENT TIME.
*****************************************************************************************
 30   CALL TIMADV
*****************************************************************************************
      THE VARIABLE 'IMEVT' INDICATES THE IMMINENT EVENT
        IMEVT =  1 IS AN OFT1 BREAK-DOWN
        IMEVT =  2 IS AN OFT2 BREAK-DOWN
        IMEVT =  3 IS A  WST1 BREAK-DOWN
        IMEVT =  4 IS A  WST2 BREAK-DOWN
        IMEVT =  5 IS A  PTT1 BREAK-DOWN
        IMEVT =  6 IS A  PTT2 BREAK-DOWN
        IMEVT =  7 IS A  PTCI BREAK-DOWN
        IMEVT =  8 IS A  PTC2 BREAK-DOWN
        IMEVT =  9 IS AN OFT1 REPAIR
        IMEVT = 10 IS AN OFT2 REPAIR
        IMEVT = 11 IS A  WST1 REPAIR
```

```
      IMEVT = 12 IS A  WST2 REPAIR
      IMEVT = 13 IS A  PTT1 REPAIR
      IMEVT = 14 IS A  PTT2 REPAIR
      IMEVT = 15 IS A  PTC1 REPAIR
      IMEVT = 16 IS A  PTC2 REPAIR
      IMEVT = 17 IS THE END OF THE SIMULATION
**********************************************************************************************
      GO TO (40,40,40,40,40,40,40,40,50,50,50,50,50,50,50,50,60),IMEVT
**********************************************************************************************
      CALL THE APPROPRIATE EVENT SUBROUTINE
**********************************************************************************************
 40   CALL BRKDN
      GO TO 30
 50   CALL REPAIR
**********************************************************************************************
        GO TO TIME ADVANCE ROUTINE TO SEE IF SIMULATION IS COMPLETE
**********************************************************************************************
        GO TO 30
**********************************************************************************************
      WHEN SIMULATION IS OVER CALL REPORT GENERATOR
**********************************************************************************************
 60   CALL RPTGEN
      STOP
      END

      SUBROUTINE INITLZ
**********************************************************************************************
      INITIALIZATION ROUTINE
**********************************************************************************************
      INTEGER F,ELAWMT(100),TNREP
      REAL MITOF1,MITOF2,MITWS1,MITWS2,MITPT1,MITPT2,MITPC1,MITPC2,
     1  MSTOF1,MSTOF2,MSTWS1,MSTWS2,MSTPT1,MSTPT2,MSTPC1,MSTPC2,
     2  FEL(17),CHKIN(8),CHKINQ(100),NORML,TREPTM(8),TTMINQ(8)
      COMMON /SIM/ MITOF1,MITOF2,MITWS1,MITWS2,MITPT1,MITPT2,MITPC1,
     1  MITPC2,MSTOF1,MSTOF2,MSTWS1,MSTWS2,MSTPT1,MSTPT2,MSTPC1,MSTPC2,
     2  SGMOF1,SGMOF2,SGMWS1,SGMWS2,SGMPT1,SGMPT2,SGMPC1,SGMPC2,NAWTMT,
     3  NMTBUS,CHKIN,CHKINQ,TREPTM,MQ,F,TNREP,NMTPER,ELAWMT,TTMINQ,
       COMMON/TIMEKP/ CLOCK,IMEVT,NUMEVS,FEL
**********************************************************************************************
      INITIALIZE SIMULATION
        1)  SET SIMULATION CLOCK TO 0
        2)  ASSUME SYSTEM IS EMPTY AND IDLE AT TIME 0
        3)  INITIALIZE CUMULATIVE STATISTICS TO 0
**********************************************************************************************
      DO 10 I = 1,8
         TTMINQ(I) = 0
         TREPTM(I) = 0
 10   CONTINUE
      CLOCK = 0.0
      IMEVT = 0
      NAWTMT = 0
      NMTBUS = 0
      TNREP = 0
      F = 0
      MQ = 0
**********************************************************************************************
      GENERATE TIMES FOR THE FIRST BREAKDOWNS AND SCHEDULE THEM IN
        FEL(1) - FEL(8), SET FEL(9) - FEL(16) TO INFINITY TO INDICATE
        THAT REPAIR IS NOT POSSIBLE WHILE THE SYSTEM IS UP, SET
        FEL(17) EQUAL TO LENGTH OF TIME SIMULATION IS TO RUN.
**********************************************************************************************
      FEL(1)   = CLOCK + EXPON(MITOF1))**(1/.84)
      FEL(2)   = CLOCK + EXPON(MITOF2))**(1/.84)
      FEL(3)   = CLOCK + EXPON(MITWS1))**(1/.7)
      FEL(4)   = CLOCK + EXPON(MITWS2))**(1/.7)
      FEL(5)   = CLOCK + EXPON(MITPT1))**(1/.77)
      FEL(6)   = CLOCK + EXPON(MITPT2))**(1/.77)
      FEL(7)   = CLOCK + EXPON(MITPC1))**(1/.73)
      FEL(8)   = CLOCK + EXPON(MITPC2))**(1/.73)
```

```
      FEL(9)    = 1.0E+30
      FEL(10)   = 1.0E+30
      FEL(11)   = 1.0E+30
      FEL(12)   = 1.0E+30
      FEL(13)   = 1.0E+30
      FEL(14)   = 1.0E+30
      FEL(15)   = 1.0E+30
      FEL(16)   = 1.0E+30
      FEL(17)   = 6000
      RETURN
      END

      SUBROUTINE TIMADV
***********************************************************************************
      TIME ADVANCE ROUTINE FINDS NEXT EVENT ON FUTURE EVENT LIST
        AND ADVANCES CLOCK
***********************************************************************************
      INTEGER F,ELAWMT(100),TNREP
      REAL MITOF1,MITOF2,MITWS1,MITWS2,MITPT1,MITPT2,MITPC1,MITPC2,
     1 MSTOF1,MSTOF2,MSTWS1,MSTWS2,MSTPT1,MSTPT2,MSTPC1,MSTPC2,
     2 FEL(17),CHKIN(8),CHKINQ(100),NORML,TREPTM(8),TTMINQ(8)
      COMMON /SIM/ MITOF1,MITOF2,MITWS1,MITWS2,MITPT1,MITPT2,MITPCl,
     1 MITPC2,MSTOF1,MSTOF2,MSTWSl,MSTWS2,MSTPT1,MSTPT2,MSTPC1,MSTPC2,
     2 SGMOF1,SGMOF2,SGMWS1,SGMWS2,SGMPT1,SGMPT2,SGMPC1,SGMPC2,NAWTMT,
     3 NMTBUS,CHKIN,CHKINQ,TREPTM,MQ,F,TNREP,NMTPER,ELAWMT,TTMINQ
      COMMON /TIMEKP/ CLOCK,IMEVT,NUMEVS,FEL
      FMIN = 1.0E+29
      IMEVT = 0
***********************************************************************************
      SEARCH THE FUTURE EVENTS LIST FOR THE NEXT EVENT
***********************************************************************************
      DO 30 I=l,NUMEVS
         IF (FEL(I).GE.FMIN) GO TO 30
            FMIN = FEL(I)
         IMEVT = I
 30      CONTINUE
         IF (IMEVT.GT.0) GO TO 50
***********************************************************************************
      ERROR CONDITION - FUTURE EVENTS LIST IS EMPTY
***********************************************************************************
 40   FORMAT(2X, '***** FUTURE EVENTS LIST EMPTY*****',
     C/,1X,'***** SIMULATION CANNOT CONTINUE *****')
      CALL RPTGEN
      STOP
***********************************************************************************
      ADVANCE SIMULATION CLOCK
        NEXT EVENT IS TYPE 'IMEVT' WHICH WILL OCCUR AT TIME FEL(IMEVT)
***********************************************************************************
 50   CLOCK = FEL(IMEVT)
      RETURN
      END

      SUBROUTINE BRKDN
***********************************************************************************
      BREAK-DOWN EVENT ROUTINE - SCHEDULES THE BREAK-DOWN EVENT
       MAINTENANCE, OR TO A QUEUE AWAITING MAINTENANCE, COMPUTE
       MAINTENANCE TIME AND UPDATE STATISTICS.
***********************************************************************************
      INTEGER F,ELAWMT(100),TNREP
      REAL MITOF1,MITOF2,MITWS1,MITWS2,MITPT1,MITPT2,MITPC1,MITPC2,
     1 MSTOF1,MSTOF2,MSTWS1,MSTWS2,MSTPT1,MSTPT2,MSTPC1,MSTPC2,
     2 FEL(17),CHKIN(8),CHKINQ(100),NORML,TREPTM(8),TTMINQ(8)
      COMMON /SIM/ MITOF1,MITOF2,MITWS1,MITWS2,MITPT1,MITPT2,MITPCl,
     1 MITPC2,MSTOF1,MSTOF2,MSTWSl,MSTWS2,MSTPT1,MSTPT2,MSTPC1,MSTPC2,
     2 SGMOF1,SGMOF2,SGMWS1,SGMWS2,SGMPT1,SGMPT2,SGMPC1,SGMPC2,NAWTMT,
     3 NMTBUS,CHKIN,CHKINQ,TREPTM,MQ,F,TNREP,NMTPER,ELAWMT,TTMINQ
      COMMON /TIMEKP/ CLOCK,IMEVT,NUMEVS,FEL
      SVT = 0
***********************************************************************************
```

```
     SCHEDULE APPROPRIATE TYPE OF BREAK-DOWN
       20 FOR   OFT1 BREAK-DOWN
       30 FOR OFT2  BREAK-DOWN
       40 FOR   WST1 BREAK-DOWN
       50 FOR WST2  BREAK-DOWN
       60 FOR PTT1  BREAK-DOWN
       70 FOR PTT2  BREAK-DOWN
       80 FOR PTC1 BREAK-DOWN
       90 FOR PTC2  BREAK-DOWN
*******************************************************************************************
     GO TO (20,30,40,50,60,70,80,90),IMEVT
*******************************************************************************************
     DETERMINE IF MAINTENANCE PERSONNEL ARE ALL BUSY
*******************************************************************************************
20   IF(NMTBUS.GE.NMTPER)  GO TO 25
*******************************************************************************************
     MAINTENANCE PERSONNEL ARE AVAILABLE, UPDATE SYSTEM STATE AND
       COMPUTE SERVICE TIME FOR REPAIR
*******************************************************************************************
     NMTBUS = NMTBUS + 1
     CHKIN(l) = CLOCK
     SVT = NORML(MSTOF1,SGMOF1)
     FEL(9) = CLOCK + SVT**(1/.33)
     FEL(1) = 1.0E+30
     GO TO 26
*******************************************************************************************
     MAINTENANCE PERSONNEL NOT AVAILABLE, PUT REPAIR IN THE Q
       UPDATE STATISTICS
*******************************************************************************************
25   NAWTMT = NAWTMT + 1
     IF (NAWTMT.GT.MQ) MQ = NAWTMT
     IF (NAWTMT.GT.99) GO TO 100
     CHKINQ(NAWTMT) = CLOCK
     ELAWMT(NAWTMT) = 1
     FEL(1) = 1.0E+30
26   RETURN
*******************************************************************************************
     DETERMINE IF MAINTENANCE PERSONNEL ARE ALL BUSY
*******************************************************************************************
30   IF(NMTBUS.GE.NMTPER) GO TO 35
*******************************************************************************************

     MAINTENANCE PERSONNEL ARE AVAILABLE, UPDATE SYSTEM STATE AND
       COMPUTE SERVICE TIME FOR REPAIR
*******************************************************************************************
     NMTBUS = NMTBUS + 1
     CHKIN(2) = CLOCK
     SVT = NORML(MSTOF2,SGMOF2)
     FEL(10) = CLOCK + EXP(SVT)
     FEL(2) = 1.0E+30
     GO TO 36
*******************************************************************************************
     MAINTENANCE PERSONNEL NOT AVAILABLE, PUT REPAIR IN THE Q
       UPDATE STATISTICS
*******************************************************************************************
35   NAWTMT = NAWTMT + 1
     IF (NAWTMT.GT.MQ) MQ = NAWTMT
     IF (NAWTMT.GT.99) GO TO 100
     CHKINQ(NAWTMT) = CLOCK
     ELAWMT(NAWTMT) = 2
     FEL(2) = 1.0E+30
36   RETURN
*******************************************************************************************
     DETERMINE IF MAINTENANCE PERSONNEL ARE ALL BUSY
*******************************************************************************************
40   IF(NMTBUS.GE.NMTPER) GO TO 45
*******************************************************************************************
     MAINTENANCE PERSONNEL ARE AVAILABLE, UPDATE SYSTEM STATE AND
       COMPUTE SERVICE TIME FOR REPAIR
```

```
*******************************************************************************************
      NMTBUS = NMTBUS + 1
      CHKIN(3) = CLOCK
      SVT = NORML(MSTWS1,SGMWS1)
      FEL(11) = CLOCK + SVT**(1/.32)
      FEL(3) = 1.0E+30
      GO TO 46
*******************************************************************************************
      MAINTENANCE PERSONNEL NOT AVAILABLE, PUT REPAIR IN THE Q
        UPDATE STATISTICS
*******************************************************************************************

 45   NAWTMT = NAWTMT + 1
      IF (NAWTMT.GT.MQ) MQ = NAWTMT
      IF (NAWTMT.GT.99) GO TO 100
      CHKINQ(NAWTMT) = CLOCK
      ELAWMT(NAWTMT) = 3
      FEL(3) = 1.0E+30
 46   RETURN
*******************************************************************************************
      DETERMINE IF MAINTENANCE PERSONNEL ARE ALL BUSY
*******************************************************************************************
 50   IF(NMTBUS.GE.NMTPER) GO TO 55
*******************************************************************************************
      MAINTENANCE PERSONNEL ARE AVAILABLE, UPDATE SYSTEM STATE AND
        COMPUTE SERVICE TIME FOR REPAIR
*******************************************************************************************
      NMTBUS = NMTBUS + 1
      CHKIN(4) = CLOCK
      SVT = NORML(MSTWS2,SGMWS2)
      FEL(12) = CLOCK + SVT**(1/.32)
      FEL(4) = 1.0E+30
      GO TO 56
*******************************************************************************************
      MAINTENANCE PERSONNEL NOT AVAILABLE, PUT REPAIR IN THE Q
        UPDATE STATISTICS
*******************************************************************************************
 55   NAWTMT = NAWTMT + 1
      IF (NAWTMT.GT.MQ) MQ = NAWTMT
      IF (NAWTMT.GT.99) GO TO 100
      CHKINQ(NAWTMT) = CLOCK
      ELAWMT(NAWTMT) = 4
      FEL(4) = 1.0E+30
 56   RETURN
*******************************************************************************************
      DETERMINE IF MAINTENANCE PERSONNEL ARE ALL BUSY
*******************************************************************************************
 60   IF(NMTBUS.GE.NMTPER) GO TO 65
*******************************************************************************************
      MAINTENANCE PERSONNEL ARE AVAILABLE, UPDATE SYSTEM STATE AND
        COMPUTE SERVICE TIME FOR REPAIR
*******************************************************************************************
      NMTBUS = NMTBUS + 1
      CHKIN(5) = CLOCK
      SVT = NORML(MSTPT1,SGMPT1)
      FEL(13) = CLOCK + SVT**(1/.5)
      FEL(5) = 1.0E+30
      GO TO 66
*******************************************************************************************
      MAINTENANCE PERSONNEL NOT AVAILABLE, PUT REPAIR IN THE Q
        UPDATE STATISTICS
*******************************************************************************************
 65   NAWTMT = NAWTMT + 1
      IF (NAWTMT.GT.MQ) MQ = NAWTMT
      IF (NAWTMT.GT.99) GO TO 100
      CHKINQ(NAWTMT) = CLOCK
      ELAWMT(NAWTMT) = 5
      FEL(5) = 1.0E+30
 66   RETURN
```

```
***********************************************************************************************
      DETERMINE IF MAINTENANCE PERSONNEL ARE ALL BUSY
***********************************************************************************************
 70   IF(NMTBUS.GE.NMTPER) GO TO 75
***********************************************************************************************
      MAINTENANCE PERSONNEL ARE AVAILABLE, UPDATE SYSTEM STATE AND
        COMPUTE SERVICE TIME FOR REPAIR
***********************************************************************************************
      NMTBUS = NMTBUS + 1
      CHKIN(6) = CLOCK
      SVT = NORML(MSTPT2,SGMPT2)
      FEL(14) = CLOCK + SVT**(1/.5)
      FEL(6) = 1.0E+30
      GO TO 76
***********************************************************************************************
      MAINTENANCE PERSONNEL NOT AVAILABLE, PUT REPAIR IN THE Q
        UPDATE STATISTICS
***********************************************************************************************
 75   NAWTMT = NAWTMT + 1
      IF (NAWTMT.GT.MQ) MQ = NAWTMT
      IF (NAWTMT.GT.99) GO TO 100
      CHKINQ(NAWTMT) = CLOCK
      ELAWMT(NAWTMT) = 6
      FEL(6) = 1.0E+30
 76   RETURN
***********************************************************************************************
      DETERMINE IF MAINTENANCE PERSONNEL ARE ALL BUSY
***********************************************************************************************
 80   IF(NMTBUS.GE.NMTPER) GO TO 85
***********************************************************************************************
      MAINTENANCE PERSONNEL ARE AVAILABLE, UPDATE SYSTEM STATE
        COMPUTE SERVICE TIME FOR REPAIR
***********************************************************************************************
      NMTBUS = NMTBUS + 1
      CHKIN(7) = CLOCK
      SVT = NORML(MSTPC1,SGMPC1)
      FEL(15) = CLOCK + SVT**(1/.14)
      FEL(7) = 1.0E+30
      GO TO 86
***********************************************************************************************
      MAINTENANCE PERSONNEL NOT AVAILABLE, PUT REPAIR IN THE Q
        UPDATE STATISTICS
***********************************************************************************************
 85   NAWTMT = NAWTMT + 1
      IF (NAWTMT.GT.MQ) MQ = NAWTMT
      IF (NAWTMT.GT.99) GO TO 100
      CHKINQ(NAWTMT) = CLOCK
      ELAWMT(NAWTMT) = 7
      FEL(7) = 1.0E+30
 86   RETURN
***********************************************************************************************
      DETERMINE IF MAINTENANCE PERSONNEL ARE ALL BUSY
***********************************************************************************************
 90   IF(NMTBUS.GE.NMTPER) GO TO 95
***********************************************************************************************
      MAINTENANCE PERSONNEL ARE AVAILABLE, UPDATE SYSTEM STATE AND
        COMPUTE SERVICE TIME FOR REPAIR
***********************************************************************************************
      NMTBUS = NMTBUS + 1
      CHKIN(8) = CLOCK
      SVT = NORML(MSTPC2,SGMPC2)
      FEL(16) = CLOCK + SVT**(1/.14)
      FEL(8) = 1.0E+30
      GO TO 96
***********************************************************************************************
      MAINTENANCE PERSONNEL NOT AVAILABLE, PUT REPAIR IN THE Q
        UPDATE STATISTICS
***********************************************************************************************
```

```
95    NAWTMT = NAWTMT + 1
      IF (NAWTMT.GT.MQ) MQ = NAWTMT
      IF (NAWTMT.GT.99) GO TO 100
      CHKINQ(NAWTMT) = CLOCK
      ELAWMT(NAWTMT) = 8
      FEL(8) = 1.0E+30
96    RETURN
100   WRITE(6,105)
105   FORMAT(3X,'***** TO MANY AWAITING MAINTENANCE ACTIONS *****',/,
     C1X,'***** INCREASE NUMBER OF MAINTENANCE PERSONNEL *****')
      CALL RPTGEN
      STOP
      END

      SUBROUTINE REPAIR
****************************************************************************************
      REPAIR EVENT ROUTINE: NOTE REPAIR COMPLETE, TAKE NLXT
        BREAK-DOWN EVENT OUT OF THE QUEUE, COMPUTE REPAIR TIME,
        SCHEDULE REPAIR EVENT.
****************************************************************************************
      INTEGER F,ELAWMT(100),TNREP
      REAL MITOF1,MITOF2,MITWS1,MITWS2,MITPT1,MITPT2,MITPC1,MITPC2,
     1  MSTOF1,MSTOF2,MSTWS1,MSTWS2,MSTPT1,MSTPT2,MSTPC1,MSTPC2,
     2  FEL(17),CHKIN(8),CHKINQ(100),NORML,TREPTM(8),TTMINQ(8)
      COMMON /SIM/ MITOF1,MITOF2,MITWS1,MITWS2,MITPT1,MITPT2,MITPCl,
     1  MITPC2,MSTOF1,MSTOF2,MSTWSl,MSTWS2,MSTPT1,MSTPT2,MSTPC1,MSTPC2,
     2  SGMOF1,SGMOF2,SGMWS1,SGMWS2,SGMPT1,SGMPT2,SGMPC1,SGMPC2,NAWTMT,
     3  NMTBUS,CHKIN,CHKINQ,TREPTM,MQ,F,TNREP,NMTPER,ELAWMT,TTMINQ
      COMMON /TIMEKP/ CLOCK,IMEVT,NUMEVS,FEL
****************************************************************************************
      SCHEDULE APPROPRIATE TYPE OF REPAIR
        20 - OFT1 REPAIR
        30 - OFT2 REPAIR
        40 - WST1 REPAIR
        50 - WST2 REPAIR
        60 - PTT1 REPAIR
        70 - PTT2 REPAIR
        80 - PTC1 REPAIR
        90 - PTC2 REPAIR
****************************************************************************************
      GO TO (20,30,40,50,60,70,80,90),IMEVT-8
****************************************************************************************
      UPDATE CUMULATIVE STATISTICS
****************************************************************************************
20    TREPTM(1) = TREPTM(1) + (CLOCK - CHKIN(l))
      TNREP = TNREP + 1
      IF((CLOCK - CHKIN(1)).GE.3.0) F = F + 1
****************************************************************************************
      CHECK CONDITION OF QUEUE
****************************************************************************************
      IF(NAWTMT).GE.1.0) GO TO 25
****************************************************************************************
      QUEUE EMPTY, SERVICE MAN BECOMES IDLE, NEXT DEPARTURE TIME
        EQUAL TO INFINITY
****************************************************************************************
      NMTBUS = NMTBUS - 1
      FEL(9) = 1.0E+30
      GO TO 28
****************************************************************************************
      QUEUE CONTAINS AT LEAST ONE MORE MAINTENANCE ACTION, START WORK ON
       THE NEXT MAINTENANCE ACTION IN THE Q, AND MOVE-THE REST UP ONE
       POSITION IN THE STACK
****************************************************************************************
25    GO TO (26,36,46,56,66,76,86,96),ELAWMT(l)
26    CHKIN(1) = CHKINQ(1)
      TTMINQ(1) = TTMINQ(1) + (CLOCK - CHKINQ(l))
      DO 27 I = 1,NAWTMT - 1
        CHKINQ(I) = CHKINQ(I+l)
        ELAWMT(I) = ELAWMT(I+l)
```

```
 27   CONTINUE
************************************************************************************
      UPDATE LENGTH OF QUEUE
************************************************************************************
      NAWTMT = NAWTMT - 1
      SVT = NORML(MSTOF1,SGMOF1)
      FEL(9) = CLOCK + SVT**(l/.33)
      RETURN
************************************************************************************
      SCHEDULE NEXT BREAK-DOWN FOR OFT1
************************************************************************************
 28   IAT = EXPON(MITOF1)
      FEL(1) = CLOCK + IAT**(l/.84)
      RETURN
************************************************************************************
      UPDATE CUMULATIVE STATISTICS
************************************************************************************
 30   TREPTM(2) = TREPTM(2) + (CLOCK - CHKIN(2))
      TNREP = TNREP + 1
      IF((CLOCK - CHKIN(2)).GE.3.0) F = F + 1
************************************************************************************
     CHECK CONDITION OF QUEUE
************************************************************************************
      IF(NAWTMT.GE.1.0) GO TO 35
************************************************************************************
      QUEUE EMPTY, SERVICE MAN BECOMES IDLE, NEXT DEPARTURE TIME
        EQUAL TO INFINITY
************************************************************************************
        NMTBUS = NMTBUS - 1    FEL(10) = 1.0E+30
      GO TO 38
************************************************************************************

      QUEUE CONTAINS AT LEAST ONE MORE MAINTENANCE ACTION, START WORK
        ON THE NEXT MAINTENANCE ACTION IN THE Q, AND MOVE ONE THE REST
        UP ONE POSITION IN THE STACK
************************************************************************************
 35   GO TO (26,36,46,56,66,76,86,96),ELAWMT(l)
 36   CHKIN(2) = CHKINQ(1)
      TTMINQ(2) = TTMINQ(2) + (CLOCK - CHKINQ(l))
      DO 37, I = 1,NAWTMT - 1
        CHKINQ(I) = CHKINQ(I+l)
        ELAWMT(I) = ELAWMT(I+l)
 37   CONTINUE
************************************************************************************
      UPDATE LENGTH OF QUEUE
************************************************************************************
      NAWTMT = NAWTMT - 1
      SVT = NORML(MSTOF2,SGMOF2)
      FEL(10) = CLOCK + SVT**(l/.33)
      RETURN
************************************************************************************
      SCHEDULE NEXT BREAK-DOWN FOR
************************************************************************************
 38   IAT = EXPON(MITOF2)
      FEL(2) = CLOCK + IAT**(l/.84)
      RETURN
************************************************************************************
      UPDATE CUMULATIVE STATISTICS
************************************************************************************
 40   TREPTM(3) = TREPTM(3) + (CLOCK - CHKIN(3))
      TNREP = TNREP + 1
      IF((CLOCK - CHKIN(3)).GE.3.0) F = F + 1
************************************************************************************
      CHECK CONDITION OF QUEUE
************************************************************************************
      IF(NAWTMT).GE.1.0) GO TO 45
************************************************************************************
      QUEUE EMPTY, SERVICE MAN BECOMES IDLE, NEXT DEPARTURE TIME
        EQUAL TO INFINITY
```

```
*****************************************************************************************
      NMTBUS = NMTBUS - 1
      FEL(11) = 1.0E+30
      GO TO 48
*****************************************************************************************
      QUEUE CONTAINS AT LEAST ONE MORE MAINTENANCE ACTION, START WORK ON
       THE NEXT MAINTENANCE ACTION IN THE Q, AND MOVE-THE REST UP ONE
       POSITION IN THE STACK
*****************************************************************************************
 45   GO TO (26,36,46,56,66,76,86,96),ELAWMT(l)
 46   CHKIN(3) = CHKINQ(1)
      TTMINQ(3) = TTMINQ(3) + (CLOCK - CHKINQ(l))
      DO 47 I = 1,NAWTMT - 1
        CHKINQ(I) = CHKINQ(I+l)
        ELAWMT(I) = ELAWMT(I+l)
 47   CONTINUE
*****************************************************************************************
      UPDATE LENGTH OF QUEUE
*****************************************************************************************
      NAWTMT = NAWTMT - 1
      SVT = NORML(MSTWS1,SGMWS1)
      FEL(11) = CLOCK + SVT**(l/.32)
      RETURN
*****************************************************************************************
      SCHEDULE NEXT BREAK-DOWN FOR WST1

*****************************************************************************************
 48   IAT = EXPON(MITWS1)
      FEL(3) = CLOCK + IAT**(l/.7)
      RETURN
*****************************************************************************************
      UPDATE CUMULATIVE STATISTICS
*****************************************************************************************
 50   TREPTM(4) = TREPTM(4) + (CLOCK - CHKIN(4))
      TNREP = TNREP + 1
      IF((CLOCK - CHKIN(4)).GE.3.0) F = F + 1
*****************************************************************************************
     CHECK CONDITION OF QUEUE
*****************************************************************************************
      IF(NAWTMT.GE.1.0) GO TO 55
*****************************************************************************************
      QUEUE EMPTY, SERVICE MAN BECOMES IDLE, NEXT DEPARTURE TIME
        EQUAL TO INFINITY
*****************************************************************************************
       NMTBUS = NMTBUS - 1
      FEL(12) = 1.0E+30
      GO TO 58
*****************************************************************************************
      QUEUE CONTAINS AT LEAST ONE MORE MAINTENANCE ACTION, START WORK
        ON THE NEXT MAINTENANCE ACTION IN THE Q, AND MOVE ONE THE REST
        UP ONE POSITION IN THE STACK
*****************************************************************************************
 55   GO TO (26,36,46,56,66,76,86,96),ELAWMT(l)
 56   CHKIN(4) = CHKINQ(1)
      TTMINQ(4) = TTMINQ(4) + (CLOCK - CHKINQ(l))
      DO 57, I = 1,NAWTMT - 1
        CHKINQ(I) = CHKINQ(I+l)
        ELAWMT(I) = ELAWMT(I+l)
 57   CONTINUE
*****************************************************************************************
      UPDATE LENGTH OF QUEUE
*****************************************************************************************
      NAWTMT = NAWTMT - 1
      SVT = NORML(MSTWS2,SGMWS2)
      FEL(12) = CLOCK + SVT**(l/.32)
      RETURN
*****************************************************************************************
      SCHEDULE NEXT BREAK-DOWN FOR WST2
*****************************************************************************************
```

```
 58   IAT = EXPON(MITWS2)
      FEL(4) = CLOCK + IAT**(l/.7)
      RETURN
*******************************************************************************************
      UPDATE CUMULATIVE STATISTICS
*******************************************************************************************
 60   TREPTM(5) = TREPTM(5) + (CLOCK - CHKIN(5))
      TNREP = TNREP + 1
      IF((CLOCK - CHKIN(5)).GE.3.0) F = F + 1
*******************************************************************************************
      CHECK CONDITION OF QUEUE
*******************************************************************************************
      IF(NAWTMT).GE.1.0) GO TO 65
*******************************************************************************************
      QUEUE EMPTY, SERVICE MAN BECOMES IDLE, NEXT DEPARTURE TIME
        EQUAL TO INFINITY
*******************************************************************************************
      NMTBUS = NMTBUS - 1
      FEL(13) = 1.0E+30
      GO TO 68
*******************************************************************************************
      QUEUE CONTAINS AT LEAST ONE MORE MAINTENANCE ACTION, START WORK ON
        THE NEXT  MAINTENANCE ACTION IN THE Q, AND MOVE-THE REST UP ONE
        POSITION IN THE STACK
*******************************************************************************************
 65   GO TO (26,36,46,56,66,76,86,96),ELAWMT(l)
 66   CHKIN(5) = CHKINQ(1)
      TTMINQ(5) = TTMINQ(5) + (CLOCK - CHKINQ(l))
      DO 67 I = 1,NAWTMT - 1
        CHKINQ(I) = CHKINQ(I+l)
        ELAWMT(I) = ELAWMT(I+l)
 67   CONTINUE
*******************************************************************************************
      UPDATE LENGTH OF QUEUE
*******************************************************************************************
      NAWTMT = NAWTMT - 1
      SVT = NORML(MSTPT1,SGMPT1)
      FEL(13) = CLOCK + SVT**(l/.5)
      RETURN
*******************************************************************************************
      SCHEDULE NEXT BREAK-DOWN FOR PTT1
*******************************************************************************************
 68   IAT = EXPON(MITPT1)
      FEL(5) = CLOCK + IAT**(l/.77)
      RETURN
*******************************************************************************************
      UPDATE CUMULATIVE STATISTICS
*******************************************************************************************
 70   TREPTM(6) = TREPTM(6) + (CLOCK - CHKIN(6))
      TNREP = TNREP + 1
      IF((CLOCK - CHKIN(6)).GE.3.0) F = F + 1
*******************************************************************************************
      CHECK CONDITION OF QUEUE
*******************************************************************************************
      IF(NAWTMT.GE.1.0) GO TO 75
*******************************************************************************************
      QUEUE EMPTY, SERVICE MAN BECOMES IDLE, NEXT DEPARTURE TIME
        EQUAL TO INFINITY
*******************************************************************************************
     FEL(14) = 1.0E+30
      GO TO 78
*******************************************************************************************
      QUEUE CONTAINS AT LEAST ONE MORE MAINTENANCE ACTION, START WORK
        ON THE NEXT MAINTENANCE ACTION IN THE Q, AND MOVE ONE THE REST
        UP ONE POSITION IN THE STACK
*******************************************************************************************
 75   GO TO (26,36,46,56,66,76,86,96),ELAWMT(l)
 76   CHKIN(6) = CHKINQ(1)
      TTMINQ(6) = TTMINQ(6) + (CLOCK - CHKINQ(l))
```

```
      DO 77, I = 1,NAWTMT - 1
        CHKINQ(I) = CHKINQ(I+l)
        ELAWMT(I) = ELAWMT(I+l)
 77   CONTINUE
*************************************************************************************
      UPDATE LENGTH OF QUEUE
*************************************************************************************
      NAWTMT = NAWTMT - 1
      SVT = NORML(MSTPT2,SGMPT2)
      FEL(14) = CLOCK + SVT**(l/.5)
      RETURN
*************************************************************************************
      SCHEDULE NEXT BREAK-DOWN FOR PTT2
*************************************************************************************
 78   IAT = EXPON(MITPT2)
      FEL(6) = CLOCK + IAT**(l/.77)
      RETURN
*************************************************************************************
      UPDATE CUMULATIVE STATISTICS
*************************************************************************************
 80   TREPTM(7) = TREPTM(7) + (CLOCK - CHKIN(7))
      TNREP = TNREP + 1
      IF((CLOCK - CHKIN(7)).GE.3.0) F = F + 1
*************************************************************************************
      CHECK CONDITION OF QUEUE
*************************************************************************************
      IF(NAWTMT).GE.1.0) GO TO 85
*************************************************************************************
      QUEUE EMPTY, SERVICE MAN BECOMES IDLE, NEXT DEPARTURE TIME
        EQUAL TO INFINITY
*************************************************************************************
      NMTBUS = NMTBUS - 1
      FEL(15) = 1.0E+30
      GO TO 88
*************************************************************************************
      QUEUE CONTAINS AT LEAST ONE MORE MAINTENANCE ACTION, START WORK ON
       THE NEXT MAINTENANCE ACTION IN THE Q, AND MOVE-THE REST UP ONE
       POSITION IN THE STACK
*************************************************************************************
 85   GO TO (26,36,46,56,66,76,86,96),ELAWMT(l)
 86   CHKIN(7) = CHKINQ(1)
      TTMINQ(7) = TTMINQ(7) + (CLOCK - CHKINQ(l))
      DO 87 I = 1,NAWTMT - 1
        CHKINQ(I) = CHKINQ(I+l)
        ELAWMT(I) = ELAWMT(I+l)
 87   CONTINUE
*************************************************************************************
      UPDATE LENGTH OF QUEUE
*************************************************************************************
      NAWTMT = NAWTMT - 1
      SVT = NORML(MSTPC1,SGMPC1)
      FEL(15) = CLOCK + SVT**(l/.14)
      RETURN
*************************************************************************************
      SCHEDULE NEXT BREAK-DOWN FOR PTT1
*************************************************************************************
 88   IAT = EXPON(MITPC1)
      FEL(7) = CLOCK + IAT**(l/.73)
      RETURN
*************************************************************************************
      UPDATE CUMULATIVE STATISTICS
*************************************************************************************
 90   TREPTM(8) = TREPTM(8) + (CLOCK - CHKIN(8))
      TNREP = TNREP + 1
      IF((CLOCK - CHKIN(8)).GE.3.0) F = F + 1
*************************************************************************************
     CHECK CONDITION OF QUEUE
*************************************************************************************
      IF(NAWTMT.GE.1.0) GO TO 95
```

```
C*****************************************************************************************
C     QUEUE EMPTY, SERVICE MAN BECOMES IDLE, NEXT DEPARTURE TIME
C       EQUAL TO INFINITY
C*****************************************************************************************
      FEL(16) = 1.0E+30
      GO TO 98
C*****************************************************************************************
C     QUEUE CONTAINS AT LEAST ONE MORE MAINTENANCE ACTION, START WORK
C       ON THE NEXT MAINTENANCE ACTION IN THE Q, AND MOVE ONE THE REST
C       UP ONE POSITION IN THE STACK
C*****************************************************************************************
 95   GO TO (26,36,46,56,66,76,86,96),ELAWMT(l)
 96   CHKIN(8) = CHKINQ(1)
      TTMINQ(8) = TTMINQ(8) + (CLOCK - CHKINQ(l))
      DO 97, I = 1,NAWTMT - 1
        CHKINQ(I) = CHKINQ(I+l)
        ELAWMT(I) = ELAWMT(I+l)
 97   CONTINUE
C*****************************************************************************************
C     UPDATE LENGTH OF QUEUE
C*****************************************************************************************
      NAWTMT = NAWTMT - 1
      SVT = NORML(MSTPC2,SGMPC2)
      FEL(16) = CLOCK + SVT**(l/.14)
      RETURN
C*****************************************************************************************
C     SCHEDULE NEXT BREAK-DOWN FOR PCT2
C*****************************************************************************************
 98   IAT = EXPON(MITPC2)
      FEL(8) = CLOCK + IAT**(l/.73)
      RETURN
      END
C*****************************************************************************************
      SUBROUTINE RPTGEN
C*****************************************************************************************
C     GENERATE REPORT OF SIMULATION RESULTS
C*****************************************************************************************
      INTEGER F,ELAWMT(100),TNREP
      REAL MITOF1,MITOF2,MITWS1,MITWS2,MITPT1,MITPT2,MITPC1,MITPC2,
     1 MSTOF1,MSTOF2,MSTWS1,MSTWS2,MSTPT1,MSTPT2,MSTPC1,MSTPC2,
     2 FEL(17),CHKIN(8),CHKINQ(100),NORML,TREPTM(8),TTMINQ(8)
      COMMON /SIM/ MITOF1,MITOF2,MITWS1,MITWS2,MITPT1,MITPT2,MITPCl,
     1 MITPC2,MSTOF1,MSTOF2,MSTWSl,MSTWS2,MSTPT1,MSTPT2,MSTPC1,MSTPC2,
     2 SGMOF1,SGMOF2,SGMWS1,SGMWS2,SGMPT1,SGMPT2,SGMPC1,SGMPC2,NAWTMT,
     3 NMTBUS,CHKIN,CHKINQ,TREPTM,MQ,F,TNREP,NMTPER,ELAWMT,TTMINQ
      COMMON /TIMEKP/ CLOCK,IMEVT,NUMEVS,FEL
C*****************************************************************************************
C     COMPUTE SUMMARY STATISTICS
C*****************************************************************************************
      OPRDY1 = (1-((TTMINQ(1) + TREPTM(1)) / CLOCK)) * 100
      OPRDY2 = (1-((TTMINQ(2) + TREPTM(2)) / CLOCK)) * 100
      OPRDY3 = (1-((TTMINQ(3) + TREPTM(3)) / CLOCK)) * 100
      OPRDY4 = (1-((TTMINQ(4) + TREPTM(4)) / CLOCK)) * 100
      OPRDY5 = (1-((TTMINQ(5) + TREPTM(5)) / CLOCK)) * 100
      OPRDY6 = (1-((TTMINQ(6) + TREPTM(6)) / CLOCK)) * 100
      OPRDY7 = (1-((TTMINQ(7) + TREPTM(7)) / CLOCK)) * 100
      OPRDY8 = (1-((TTMINQ(8) + TREPTM(8)) / CLOCK)) * 100
C*****************************************************************************************
C     WRITE OUTPUT
C*****************************************************************************************
      WRITE(6,10)
 10   FORMAT(6X, 'SIMULATION MODEL OF AIRCRAFT SIMULATOR MAINTENANCE',
     1///)
        WRITE(6,11)     MITOF1,MITOF2,MITWS1,MITWS2,MITPT1,MITPT2,MITPC1,
     lMITPC2,MSTOFl,MSTOF2,MSTWS1,MSTWS2,MSTPT1,MSTPT2,MSTPCl,MSTPC2
 11   FORMAT(1X,'LAMBDA USED FOR OFT1 FAILURE TIMES              ',F10.2
     1      /1X,'LAMBDA USED FOR OFT2 FAILURE TIMES              ',F10.2
     2      /1X,'LAMBDA USED FOR WSTI FAILURE TIMES              ',F10.2
     3      /1X,'LAMBDA USED FOR WST2 FAILURE TIMES              ',F10.2
```

```
     4       /1X,'LAMBDA USED FOR PTT1 FAILURE TIMES            ',F10.2
     5       /1X,'LAMBDA USED FOR PTT2 FAILURE TIMES            ',F10.2
     6       /1X,'LAMBDA USED FOR PTCI FAILURE TIMES            ',F10.2
     7       /1X,'LAMBDA USED FOR PTC2 FAILURE TIMES            ',F10.2
     8       /1X,'MEAN SERVICE TIME FOR OFT1 REPAIR             ',F10.2
     9       /1X,'MEAN SERVICE TIME FOR OFT2 REPAIR             ',F10.2
     A       /1X,'MEAN SERVICE TIME FOR WSTI REPAIR             ',F10.2
     B       /1X,'MEAN SERVICE TIME FOR WST2 REPAIR             ',F10.2
     C       /1X,'MEAN SERVICE TIME FOR PTTI REPAIR             ',F10.2
     D       /1X,'MEAN SERVICE TIME FOR PTT2 REPAIR             ',F10.2
     E       /1X,'MEAN SERVICE TIME FOR PTC1 REPAIR             ',F10.2
     F       /1X,'MEAN SERVICE TIME FOR PTC2 REPAIR             ',F10.2)
      WRITE(6,12) SGMOF1,SGMOF2,SGMWS1,SGMWS2,SGMPTI,SGMPT2,SGMPC1,
     1SGMPC2,NMTPER
 12   FORMAT(1X,'STANDARD DEVIATION FOR OFT1 REPAIR             ',F10.2
     1       /1X,'STANDARD DEVIATION FOR OFT2 REPAIR            ',F10.2
     2       /1X,'STANDARD DEVIATION FOR WST1 REPAIR            ',F10.2
     3       /1X,'STANDARD DEVIATION FOR WST2 REPAIR            ',F10.2
     4       /1X,'STANDARD DEVIATION FOR PTT1 REPAIR            ',F10.2
     5       /1X,'STANDARD DEVIATION FOR PTT2 REPAIR            ',F10.2
     6       /1X,'STANDARD DEVIATION FOR PTC1 REPAIR            ',F10.2
     7       /1X,'STANDARD DEVIATION FOR PTC2 REPAIR            ',F10.2
     8      /1X,'NUMBER OF MAINTENANCE PERSONNEL               ',I10.0)
      WRITE(6,13)MQ,OPRDYI,OPRDY2,OPRDY3,OPRDY4,OPRDY5,OPRDY6,
        1OPRDY7,OPRDY8,F,CLOCK,TNREP
 13   FORMAT(1X,'MAXIMUM QUEUE LENGTH                           ',I10.0
     2  /1X,'OFT1 OPERATIONAL READY RATE                  ',F11.0,'%'
     3  /1X,'OFT2 OPERATIONAL READY RATE                  ',F11.0,'%'
     4  /1X,'WST1 OPERATIONAL READY RATE                  ',F11.0,'%'
     5  /1X,'WST2 OPERATIONAL READY RATE                  ',F11.0,'%'
     6  /1X,'PTT1 OPERATIONAL READY RATE                  ',F11.0,'%'
     7  /1X,'PTT2 OPERATIONAL READY RATE                  ',F11.0,'%'
     8  /1X,'PTC1 OPERATIONAL READY RATE                  ',F11.0,'%'
     9  /1X,'PTC2 OPERATIONAL READY RATE                  ',F11.0,'%'
     A  /1X,'NUMBER OF MAINTENANCE ACTIONS THAT TOOK        ',
     B  /1X,'MORE THAN THREE (3) HOURS TO CORRECT         ',F10.2
     C  /1X,'SIMULATION RUN TIME                          ',F10.2
     D /1X,'NUMBER OF MAINTENANCE ACTIONS DURING SIMULATION 'I10.0)
      RETURN
      END
****************************************************************************************************
```

```
************************************************************************************************
                 EXPONENTIAL RANDOM VARIATE GENERATOR

     DEFINITION OF VARIABLES

       ALOG     -  Natural log function
       DSEED    -  Seed for random U(0,1) number generator
       EXPON    -  Exponential random variate
       FMEAN    -  Mean parameter of exponential distribution
       GGUBFS   -  U(0,1) random number generator
       R        -  Random number

************************************************************************************************
      FUNCTION EXPON (FMEAN)
      DOUBLE PRECISION DSEED
      DATA DSEED /43/
************************************************************************************************
      GENERATE A U(0,1) RANDOM NUMBER
************************************************************************************************
      R=GGUBFS(DSEED)
************************************************************************************************
      GENERATE AN EXPONENTIAL RANDOM VARIABLE WITH MEAN FMEAN
************************************************************************************************
      EXPON=-FMEAN*ALOG(R)
      RETURN
      END
************************************************************************************************
************************************************************************************************
                              NORMAL RANDOM VARIATE GENERATOR

     DEFINITION OF VARIABLES

       MEAN     -  Mean of normal distribution
       SIGMA    -  Standard deviation of normal distribution
       NSEED    -  Seed for U(0,1) random number generator
       NORML    -  Normal random variate
       K        -  Index to determine which method of computation to use
       PI       -  Pi
       RONE     -  A U(0,1) random number
       RTWO     -  Another U(0,1) random number
       ZONE     -  A N(0,1) random variable
       ZTWO     -  Another N(0,1) random variable

************************************************************************************************
      FUNCTION NORML (MEAN,SIGMA)
      REAL MEAN,SIGMA,NORML
      DOUBLE PRECISION NSEED
      DATA K /0/, PI /3,14159/, NSEED /58/
************************************************************************************************
      CHECK TO SEE WHICH N(0,1) RANDOM VARIABLE TO USE

************************************************************************************************
************************************************************************************************
      GENERATE TWO U(0,1) RANDOM NUMBERS
************************************************************************************************
      RONE=GGUBFS(NSEED)
      RTWO=GGUBFS(NSEED)
************************************************************************************************
      GENERATE TWO N(0,1) RANDOM VARIABLES
************************************************************************************************
      ZONE=SQRT(-2*ALOG(RONE))*COS(2*PI*RTWO)
      ZTWO=SQRT(-2*ALOG(RONE))*SIN(2*PI*RTWO)
************************************************************************************************
      COMPUTE NORMAL RANDOM VARIATE WITH PARAMETERS (MEAN,SIGMA)
        FOR MEAN AND STANDARD DEVIATION
************************************************************************************************
      NORML=ZONE*SIGMA+MEAN
      K=1
      RETURN
```

```
*****************************************************************************************************
     COMPUTE NORMAL RANDOM VARIATE WITH PARAMETERS (MEAN,SIGMA**2)
       FOR MEAN AND VARIANCE
*****************************************************************************************************
 10  NORML=ZTWO*SIGMA+MEAN
     K=0
     RETURN
     END
*****************************************************************************************************
```

```
C     IMSL ROUTINE NAME    - GGUBFS                                          GGUN0010
C                                                                            GGUN0020
C----------------------------------------------------------------------------GGUN0030
C                                                                            GGUN0040
C     COMPUTER             - IBM/SINGLE                                      GGUN0050
C                                                                            GGUN0060
C     LATEST REVISION      - JUNE 1, 1980                                    GGUN0070
C                                                                            GGUN0080
C     PURPOSE              - BASIC UNIFORM (0,1)RANDOMNUMBER                 GGUN0090
C                            GENERATOR-FUNCTION FORM OF GGUBS                GGUN0100
C                                                                            GGUN0110
C     USAGE                - FUNCTION GGUBFS (DSEED)                         GGUN0120
C                                                                            GGUN0130
C     ARGUMENTS    GGUBFS  - RESULTANT DEVIATE.                              GGUN0140
C                  DSEED   - INPUT/OUTPUT DOUBLE PRECISION VARIABLE          GGUN0150
C                              ASSIGNED AN INTEGER VALUE IN THE              GGUN0160
C                              EXCLUSIVE RANGE (l.DO, 2147483647.DO          GGUN0170
C                              DSEED IS REPLACED BY A NEW VALUE TO           GGUN0180
C                              USED IN A SUBSEQUENT CALL.                    GGUN0190
C                                                                            GGUN0200
C      PRECISION/HARDWARE  - SINGLE/ALL                                      GGUN0210
C                                                                            GGUN0220
C      REQD. IMSL ROUTINES - NONE REQUIRED                                   GGUN0230
C                                                                            GGUN0240
C      NOTATION            - INFORMATION ON SPECIAL NOTATION AND             GGUN0250
C                            CONVENTIONS IS AVAILABLE IN THE MANUAL          GGUN0260
C                              INTRODUCTION OR THROUGH IMSL ROUTINE          GGUN0270
C                              UHELP                                         GGUN0280
C                                                                            GGUN0290
C      COPYRIGHT           - 1978 BY IMSL, INC. ALL RIGHTS RESERVED          GGUN0300
C                                                                            GGUN0310
C      WARRANTY            - IMSL WARRANTS ONLY THAT IMSL TESTING            GGUN0320
C                              HAS BEEN APPLIED TO THIS CODE. NO             GGUN0330
C                              OTHER WARRANTY, EXPRESSED OR IMPLIED,         GGUN0340
C                              IS APPLICABLE.                                GGUN0350
C                                                                            GGUN0360
C----------------------------------------------------------------------------GGUN0370
C                                                                            GGUN0380
      REAL FUNCTION GGUBFS (DSEED)                                           GGUN0390
C                 SPECIFICATIONS FOR ARGUMENTS                               GGUN0400
      DOUBLE PRECISION DSEED                                                 GGUN0410
C           SPECIFICATIONS FOR LOCAL VARIABLES                               GGUN0420
      DOUBLE PRECISION D2P31M,D2P31                                          GGUN0430
C           D2P31M=(2**31) - 1                                               GGUN0440
C           D2P31 =(2**31)(OR AN ADJUSTED VALUE)                             GGUN0450
      DATA            D2P31M/2147483647.DO/                                  GGUN0460
      DATA            D2P31 /2147483648.DO/                                  GGUN0470
C                     FIRST EXECUTABLE STATEMENT                             GGUN0480
      DSEED = DMOD( 16807.DO DSEED,D2P31M)                                   GGUN0490
      GGUBFS = DSEED / D2P31                                                 GGUO0500
      RETURN                                                                 GGUN0510
      END                                                                    GGUN0520
```

APPENDIX B

SIMULATION MODEL OF AIRCRAFT SIMULATOR MAINTENANCE

LAMBDA USED FOR OFT1 FAILURE TIMES	25.80
LAMBDA USED FOR OFT2 FAILURE TIMES	25.80
LAMBDA USED FOR WST1 FAILURE TIMES	22.64
LAMBDA USED FOR WST2 FAILURE TIMES	22.64
LAMBDA USED FOR PTT1 FAILURE TIMES	28.05
LAMBDA USED FOR PTT2 FAILURE TIMES	28.05
LAMBDA USED FOR PTC1 FAILURE TIMES	21.99
LAMBDA USED FOR PTC2 FAILURE TIMES	21.99
MEAN SERVICE TIME FOR OFT1 REPAIR	1.14
MEAN SERVICE TIME FOR OFT2 REPAIR	2.08
MEAN SERVICE TIME FOR WST1 REPAIR	1.46
MEAN SERVICE TIME FOR WST2 REPAIR	1.46
MEAN SERVICE TIME FOR PTT1 REPAIR	2.23
MEAN SERVICE TIME FOR PTT2 REPAIR	2.23
MEAN SERVICE TIME FOR PTCI REPAIR	1.05
MEAN SERVICE TIME FOR PTC2 REPAIR	1.05
STANDARD DEVIATION FOR OFTI REPAIR	0.22
STANDARD DEVIATION FOR OFT2 REPAIR	0.36
STANDARD DEVIATION FOR WST1 REPAIR	0.43
STANDARD DEVIATION FOR WST2 REPAIR	0.43
STANDARD DEVIATION FOR PTT1 REPAIR	0.74
STANDARD DEVIATION FOR PTT2 REPAIR	0.74
STANDARD DEVIATION FOR PTC1 REPAIR	0.07
STANDARD DEVIATION FOR PTC2 REPAIR	0.07
NUMBER OF MAINTENANCE PERSONNEL	2
MAXIMUM QUEUE LENGTH	1
OFT1 OPERATIONAL READY RATE	97.%
OFT2 OPERATIONAL READY RATE	87.%
WST1 OPERATIONAL READY RATE	97.%
WST2 OPERATIONAL READY RATE	96.%
PTT1 OPERATIONAL READY RATE	95.%
PTT2 OPERATIONAL READY RATE	93.%
PTCI OPERATIONAL READY RATE	98.%
PTC2 OPERATIONAL READY PATE	98.%
NUMBER OF MAINTENANCE ACTIONS THAT TOOK MORE THAN FOUR (4) HOURS TO CORRECT	0.00
SIMULATION RUN TIME	6000.00
NUMBER OF MAINTENANCE ACTIONS DURING SIMULATION	550

SIMULATION MODEL OF AIRCRAFT SIMULATOR MAINTENANCE

LAMBDA USED FOR OFT1 FAILURE TIMES	25.80
LAMBDA USED FOR OFT2 FAILURE TIMES	25.80
LAMBDA USED FOR WST1 FAILURE TIMES	22.64
LAMBDA USED FOR WST2 FAILURE TIMES	22.64
LAMBDA USED FOR PTT1 FAILURE TIMES	28.05
LAMBDA USED FOR PTT2 FAILURE TIMES	28.05
LAMBDA USED FOR PTC1 FAILURE TIMES	21.99
LAMBDA USED FOR PTC2 FAILURE TIMES	21.99
MEAN SERVICE TIME FOR OFT1 REPAIR	1.14
MEAN SERVICE TIME FOR OFT2 REPAIR	2.08
MEAN SERVICE TIME FOR WST1 REPAIR	1.46
MEAN SERVICE TIME FOR WST2 REPAIR	1.46
MEAN SERVICE TIME FOR PTT1 REPAIR	2.23
MEAN SERVICE TIME FOR PTT2 REPAIR	2.23
MEAN SERVICE TIME FOR PTCI REPAIR	1.05
MEAN SERVICE TIME FOR PTC2 REPAIR	1.05
STANDARD DEVIATION FOR OFTI REPAIR	0.22
STANDARD DEVIATION FOR OFT2 REPAIR	0.36
STANDARD DEVIATION FOR WST1 REPAIR	0.43
STANDARD DEVIATION FOR WST2 REPAIR	0.43
STANDARD DEVIATION FOR PTT1 REPAIR	0.74
STANDARD DEVIATION FOR PTT2 REPAIR	0.74
STANDARD DEVIATION FOR PTC1 REPAIR	0.07
STANDARD DEVIATION FOR PTC2 REPAIR	0.07
NUMBER OF MAINTENANCE PERSONNEL	3
MAXIMUM QUEUE LENGTH	1
OFT1 OPERATIONAL READY RATE	97.%
OFT2 OPERATIONAL READY RATE	86.%
WST1 OPERATIONAL READY RATE	96.%
WST2 OPERATIONAL READY RATE	95.%
PTT1 OPERATIONAL READY RATE	94.%
PTT2 OPERATIONAL READY RATE	95.%
PTCI OPERATIONAL READY RATE	98.%
PTC2 OPERATIONAL READY PATE	98.%
NUMBER OF MAINTENANCE ACTIONS THAT TOOK MORE THAN FOUR (4) HOURS TO CORRECT	0.00
SIMULATION RUN TIME	6000.00
NUMBER OF MAINTENANCE ACTIONS DURING SIMULATION	553

SIMULATION MODEL OF AIRCRAFT SIMULATOR MAINTENANCE

LAMBDA USED FOR OFT1 FAILURE TIMES	25.80
LAMBDA USED FOR OFT2 FAILURE TIMES	25.80
LAMBDA USED FOR WST1 FAILURE TIMES	22.64
LAMBDA USED FOR WST2 FAILURE TIMES	22.64
LAMBDA USED FOR PTT1 FAILURE TIMES	28.05
LAMBDA USED FOR PTT2 FAILURE TIMES	28.05
LAMBDA USED FOR PTC1 FAILURE TIMES	21.99
LAMBDA USED FOR PTC2 FAILURE TIMES	21.99
MEAN SERVICE TIME FOR OFT1 REPAIR	1.14
MEAN SERVICE TIME FOR OFT2 REPAIR	2.08
MEAN SERVICE TIME FOR WST1 REPAIR	1.46
MEAN SERVICE TIME FOR WST2 REPAIR	1.46
MEAN SERVICE TIME FOR PTT1 REPAIR	2.23
MEAN SERVICE TIME FOR PTT2 REPAIR	2.23
MEAN SERVICE TIME FOR PTCI REPAIR	1.05
MEAN SERVICE TIME FOR PTC2 REPAIR	1.05
STANDARD DEVIATION FOR OFTI REPAIR	0.22
STANDARD DEVIATION FOR OFT2 REPAIR	0.36
STANDARD DEVIATION FOR WST1 REPAIR	0.43
STANDARD DEVIATION FOR WST2 REPAIR	0.43
STANDARD DEVIATION FOR PTT1 REPAIR	0.74
STANDARD DEVIATION FOR PTT2 REPAIR	0.74
STANDARD DEVIATION FOR PTC1 REPAIR	0.07
STANDARD DEVIATION FOR PTC2 REPAIR	0.07
NUMBER OF MAINTENANCE PERSONNEL	4
MAXIMUM QUEUE LENGTH	0
OFT1 OPERATIONAL READY RATE	97.%
OFT2 OPERATIONAL READY RATE	87.%
WST1 OPERATIONAL READY RATE	98.%
WST2 OPERATIONAL READY RATE	96.%
PTT1 OPERATIONAL READY RATE	94.%
PTT2 OPERATIONAL READY RATE	94.%
PTCI OPERATIONAL READY RATE	98.%
PTC2 OPERATIONAL READY RATE	98.%
NUMBER OF MAINTENANCE ACTIONS THAT TOOK MORE THAN FOUR (4) HOURS TO CORRECT	0.00
SIMULATION RUN TIME	6000.00
NUMBER OF MAINTENANCE ACTIONS DURING SIMULATION	552

LIST OF REFERENCES

1. Batchelder, C.A. et al, An Introduction to Equipment Cost Estimating, The Rand Corporation, Santa Monica, Ca., 1969
2. The Random House Dictionary of the English Language, Unabridged Edition, Random House, New York, NY., 1969.
3. Hoover, S.V. and Perry, R.F., Simulation A Problem-Solving Approach, Addison-Wesley Publishing Co., New York, NY.,1989

BIBLIOGRAPHY

Banks, J., Carson, J.S. II, Discrete-Event System Simulation, Prentice-Hall, Inc., Englewood Cliffs, NJ., 1984

Batchelder, C.A. et al, An Introduction to Equipment Cost Estimating, The Rand Corporation, Santa Monica, CA., 1969

Chambers, J.M. et al, Graphical Methods for Data Analysis, Duxbury Press, Boston, MA., 1983

Hoover, S.V. and Perry, R.F., Simulation A Problem-Solving Approach, Addison-Wesley Publishing Co., New York, NY.,1989

Larson, R.J., & Marx, M.L., An Introduction to Mathematical Statistics and its Applications, Prentice-Hall, Englewood Cliffs, NJ., 1986

Nyhoff, L., Leestma, S., FORTRAN 77 for Engineers and Scientists, Macmillan Publishing Co., New York, NY., 1985

Ross, S.M., Introduction to Probability Models, Academic Press, Inc., Orlando, FL., 1985

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22304-6145	2
2.	Library, Code 52 Naval Postgraduate School Monterey, California 93943-5000	2
3.	Commander, Naval Air Systems Command Aviation Training Office PMA-205 Code 205-5 Washington, D.C. 20361	2
4.	Professor D. Boger, Code AS/Bo Department of Administrative Sciences Naval Postgraduate School Monterey, California 93943-5000	1
5.	Professor W. Max Wood, Code OR/Wo Department of Operations Research Naval Postgraduate School Monterey, California 93943-5000	1
6.	Commanding Officer Naval Training Systems Center Field Engineering and Fleet Support Department Code 4 Orlando, Florida 32813-7100	1
7.	Commanding Officer Naval Training Systems Center Field Engineering and Fleet Support Department Code 432 Orlando, Florida 32813-7100	1
8.	CDR Edward M. Biggers 4884 Oakcrest Dr Fairfax, Virginia 22030	4
9.	Operations Analysis Programs, Code 30 Naval Postgraduate School Monterey, California 93943-5000	1